

(주)스타셀

# 딥셀 큐브 Notebooks 사용자 안내서

우송대학교 소프트웨어학부 시스템 사용자용

Starcell, all right reserved  
2025-10-23

# 목차

I. 개요 .....	2
1. 사전요구사항(prerequisites) .....	2
2. hosts 파일 수정 .....	2
II. 사용 방법 .....	3
1. 웹 브라우저로 접속 .....	3
2. Notebooks 사용하기 .....	4
1) Notebook 서버 만들기 .....	4
2) Jupyter Notebook에서 GPU 확인 .....	9
3) Jupyter Notebook에서 딥러닝 실행 .....	10
4) Notebook Server 정지하기 .....	10
5) Notebook Server 삭제하기 .....	11
6) Volumes 관리 .....	12
3. Contributor 관리 .....	12
III. 트러블 슈팅 .....	14

# I. 개요

## 1. 사전 요구사항(prerequisites)

딥셀큐브는 Kubernetes, Kubeflow 를 기반으로 가상의 Jupyter Notebook Server 를 생성하여 사용하는 환경을 제공합니다. 이 문서는 딥셀큐브가 제공하는 인프라가 설치된 상태에서 Jupyter Lab 환경을 사용하는 방법을 설명합니다. 이 문서의 내용을 실행하기 위해서는 다음과 같은 사항들이 필요하므로 딥셀큐브 시스템 관리자에게 문의하여 다음의 정보를 알아야 합니다.

Hosts 파일 수정 : 도메인 네임을 사용할 수 있도록 수정

접속 URL : Kubeflow central dashboard 를 접속하기 위한 주소

접속 ID/비밀번호 : Kubeflow central dashboard 에 로그인 하여 사용할 수 있는 자격과 권한을 부여하는 ID 와 비밀 번호

## 2. hosts 파일 수정

도메인 네임을 사용하기 위해서 DNS 에서 이름 매핑지 지원되지 않는 경우 시스템의 hosts 파일을 수정해서 도메인 네임을 이용하여 네트워크 접속을 할 수 있습니다. 각 시스템 별 hosts 파일의 위치는 다음과 같으므로 수정해서 사용할 수 있습니다.

LINUX, apple 등 unix 계열 : 관리자 권한 필요

/etc/hosts

Windows : 관리자 권한 필요하며, 수정된 경우 원상태로 되돌리기를 원하는지 물어봄, 되돌리면 안됨

C:\Windows\System32\drivers\etc\hosts

수정할 내용은 다음과 같습니다. 다음의 내용을 hosts 파일에 추가하고 저장합니다.

# 1. kubeflow central dashboard : https://kubeflow.deepcell.co.kr

220.90.190.241 kubeflow.deepcell.co.kr

# 2. ltb-passwd 서비스, 사용자가 자신의 딥셀큐브 ID(kubeflwo ID)의 비밀 번호를 변경할 때 접속하는 URL

220.90.190.242 userpw.deepcell.co.kr # http://userpw.deepcell.co.kr

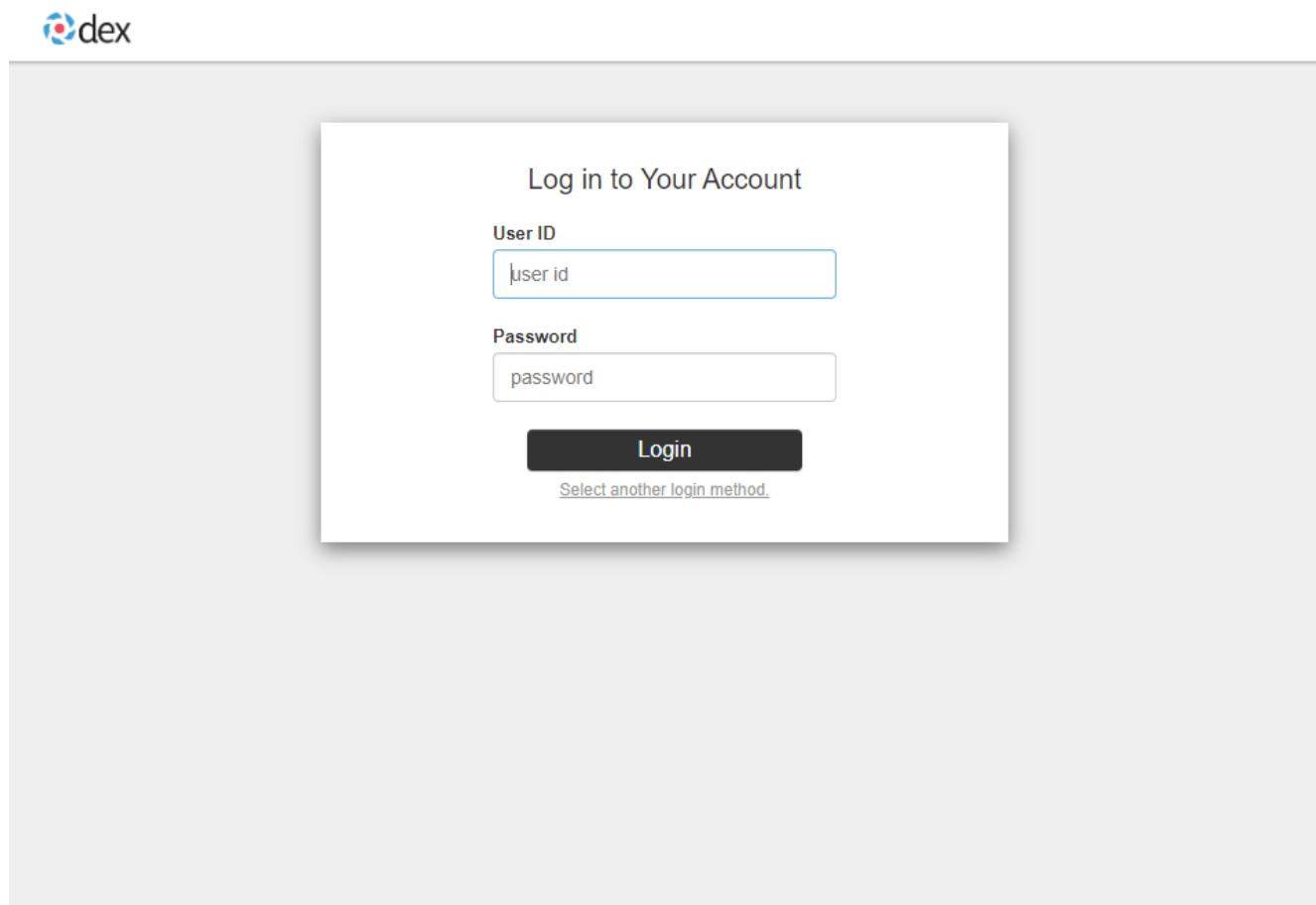
## II. 사용 방법

### 1. 웹 브라우저로 접속

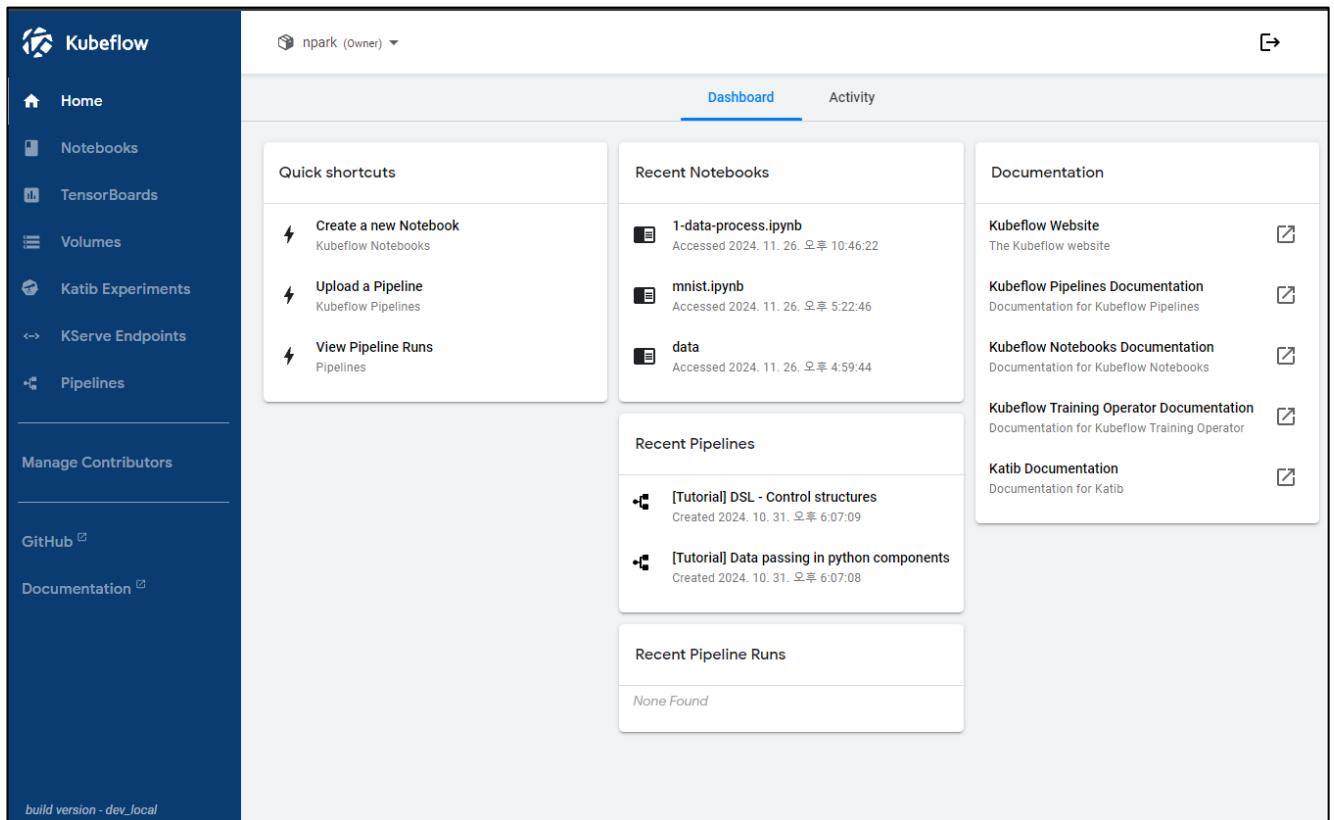
앞의 내용대로 딥셀큐브 인프라가 준비되어 있고 도메인 네임으로 네트워크 접속을 할 수 있는 준비가 되고 나면, Kubeflow central dashboard 를 사용할 수 있습니다. 다음과 같은 접속 URL 을 웹브라우저의 주소창에 입력하면 다음과 같은 로그인 화면이 나타납니다.

접속 URL : <https://kubeflow.deepcell.co.kr>

로그인 화면



로그인을 위한 User ID 와 Password 가 필요합니다. 없는 경우 관리자에게 문의하여 User ID 와 Password 를 받아서 사용하시기 바랍니다. User ID 와 Password 를 입력하여 로그인 성공하면 다음과 같은 Kubeflow Central Dashboard 화면 표시됩니다.



## 2. Notebooks 사용하기

Kubeflow central dashboard에 접속하면 Notebooks 메뉴를 이용하여 Jupyter notebook server를 만들어서 사용할 수 있습니다.

### 1) Notebook 서버 만들기

왼쪽의 메뉴에서 Notebooks 선택하면 Notebooks 화면이 표시되고 오른쪽 위의 “New Notebook” 메뉴를 클릭하여 새로운 노트북 서버를 생성할 수 있습니다.

The screenshot shows the KubeFlow interface. On the left, there is a sidebar with the following menu items:

- Home
- Notebooks (selected, highlighted with a red oval)
- TensorBoards
- Volumes
- Katib Experiments
- KServe Endpoints
- Pipelines

Below the sidebar, there is a "Manage Contributors" section and links to GitHub and Documentation.

The main content area is titled "Notebooks". It shows a table with one row of data:

Status	Name	Type	Created at	Last activity	Image	GPUs	CPUs	Memory	Actions
✓	torch-cuda-su...	Jupyter	6 days ago	-	jupyter-pytorch-...	1	2	8.0 Gi	<b>CONNECT</b>

At the top right of the main content area, there is a button labeled "+ New Notebook" with a red oval around it. The bottom right corner of the main content area has a small question mark icon.

“New Notebook”메뉴를 실행하면 다음과 같이 새로운 Notebook Server 를 생성하는 화면이 나타납니다.

Kubeflow npark (owner) ↗

### New notebook

Name 1


**JupyterLab**  
 An interactive development environment for notebooks, code, and data. Ideal for prototyping and experimentation.

**1**  
**VisualStudio Code**  
 A lightweight but powerful source code editor, redefined and optimized for building and debugging modern web and cloud applications.

**2**  
**RStudio**  
 An integrated development environment for R, a programming language for statistical computing and graphics.

Custom Notebook 3

CPU / RAM ?

Minimum CPU 4 0.5      Minimum Memory Gi 5 1

GPU Options

Number of GPUs 6 None      GPU Vendor 7

Workspace Volume

Volume that will be mounted in your home directory.

New volume 8 -workspace, Empty, 5Gi

Data Volumes 9

Additional volumes that will be mounted in your Notebook.

+ Add new volume      + Attach existing volume

Advanced Options 10

11

## <메뉴 설명>

- ① Name : 이름, 사용한 Jupyter Notebook Server 의 이름을 지정
- ② JupyterLab : 사용할 사용자 인터페이스로 JupyterLab 을 선택
- ③ Custom Notebook : 등록되어 있는 컨테이너 이미지를 선택, Notebook Server 를 생성할 때 선택된 컨테이너 이미지를 사용함, 이 이미지에는 pytorch 등과 같은 필요 소프트웨어들이 설치되어 있습니다. 기본으로 제공되는 이미지는 다음과 같음

- A. jupyter-scipy  
scipy package 가 설치된 이미지, GPU 사용 못함
- B. jupyter-pytorch-full  
CPU 용 pytorch 가 설치된 이미지, GPU 사용 못함
- C. jupyter-pytorch-cuda-full  
NVIDIA GPU 용 pytorch 가 설치된 이미지, NVIDIA GPU 사용 가능
- D. jupyter-pytorch-gaudi-full  
Intel GPU 용 pytorch 가 설치된 이미지, Intel GPU 사용 가능
- E. jupyter-tensorflow-full  
CPU 용 tensorflow 가 설치된 이미지, GPU 사용 못함
- F. jupyter-tensorflow-cuda-full  
NVIDIA GPU 용 tensorflow 가 설치된 이미지, NVIDIA GPU 사용 가능

위 이미지들을 사용하여 만든 Notebook Server 에서는 공통적으로 pip 를 이용하여 필요 패키지를 설치할 수 있음, 그러나 sudo 명령을 사용할 수 없으므로 apt(또는 apt-get) 명령으로 시스템 패키지를 설치하는 것은 안됨. sudo 명령을 사용할 수 있는 Notebook Server 가 필요한 경우 관리자에게 별도 문의 필요

- ④ CPU : 사용할 CPU 수를 설정, 자연수 또는 소수점 단위로 설정할 수 있음, 이 수는 CPU 코어의 수를 의미하며 소수점 단위로 설정할 경우 CPU 코어를 Time-Slicing 으로 분할해서 사용하는 비율을 의미함(0.5 의 경우 500mCPU 시간을 의미, 즉 하나의 코어를 1 초에 0.5 초 사용할 수 있음)
- ⑤ Minimum Memory : Notebook Server 가 시작될 때 필요한 최소한의 메모리, 시스템에 이 값 이상의 여분의 메모리가 없으면 Notebook Server 를 시작되지 않음
- ⑥ GPUs : Notebook Server 에서 사용할 GPU 의 수, 가능한 값은 1, 2, 4, 8
- ⑦ GPU Vendor : Notebook Server 가 사용할 GPU 의 제조사, NVIDIA, AMD, Intel 중 선택
- ⑧ New Volume : 사용자의 home directory 가 사용할 스토리지를 설정, 스토리지 크기 설정
- ⑨ Data Volumes : home directory 외에 추가적으로 사용할 스토리지를 설정(option)
- ⑩
- ⑪
- ⑫ Launch : 위의 설정을 완료하고 Launch 메뉴를 클릭하면 설정에 따라 새로운 Notebook Server 가 생성됨

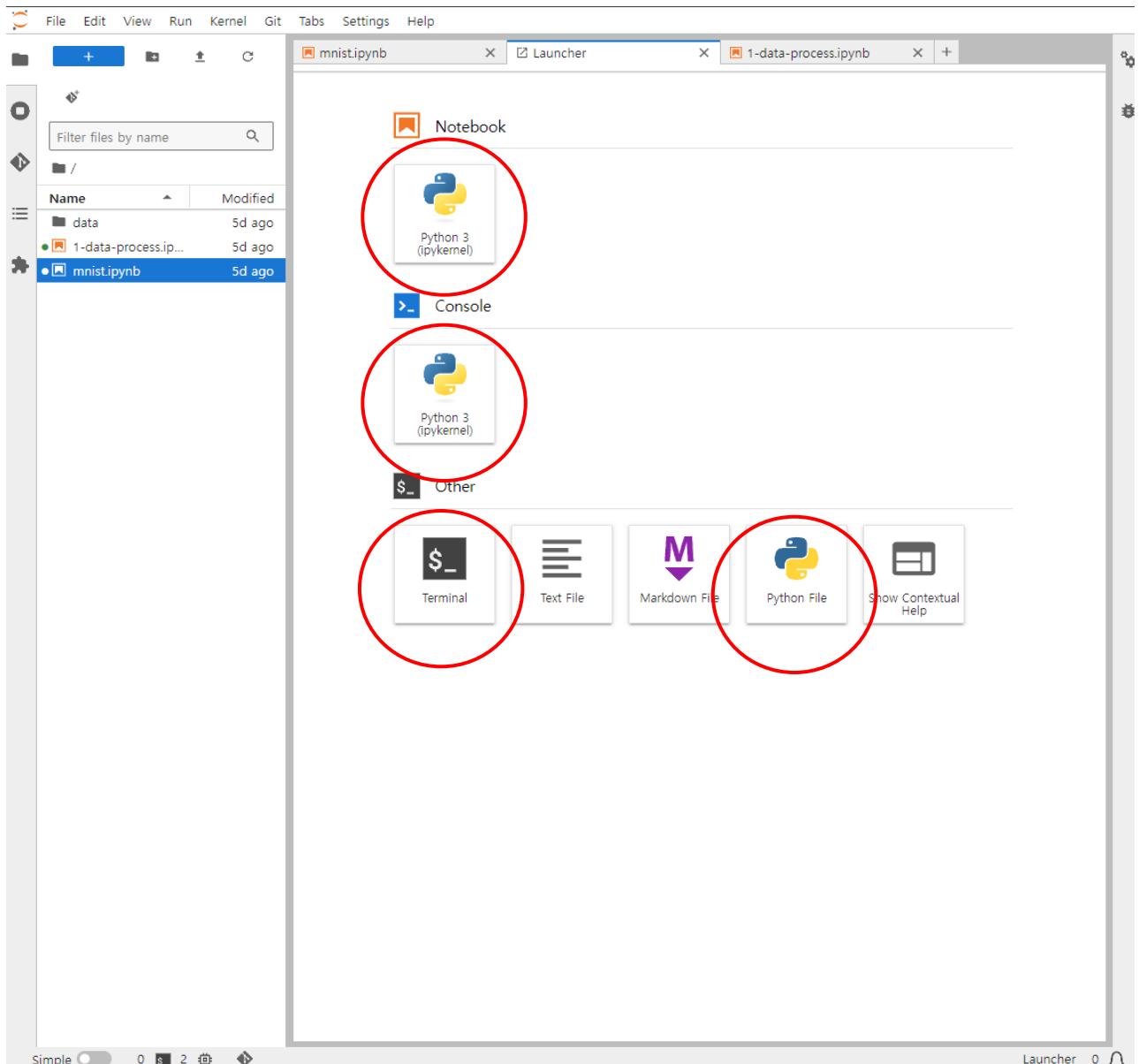
앞의 단계를 수행하고 나면 새로운 Notebook Server 가 생성됩니다. Notebooks 목록에 새로 추가한 Notebook Server 가 나타나며 왼쪽 끝에 상태 정보가 표시됩니다. 상태정보가 녹색으로 변경이 되면 Notebook Server 의 생성이 완료된 것입니다.

The screenshot shows the Kubeflow Notebooks interface. At the top, there are navigation icons and the text "Kubeflow" and "npark (Owner)". On the right, there is a "New Notebook" button and a "CONNECT" button. Below this is a search bar with the placeholder "Enter property name or value". A red circle highlights the "Status" column header. The main table has columns: Status, Name ↑, Type, Created at, Last activity, Image, GPUs, CPUs, Memory, and CONNECT. The first row shows a green checkmark icon in the Status column, indicating the notebook is active. The notebook name is "torch-cuda-sudo", type is "jupyter", created 6 days ago, last activity was at the time of creation, image is "jupyter-pytorch-cu...", and it has 1 GPU, 2 CPUs, and 8.0 GiB of memory. The "CONNECT" button is also highlighted with a red circle. A callout box labeled "상태정보" points to the green checkmark icon. The bottom of the table includes pagination controls: "Items per page: 10", "1 - 1 of 1", and navigation arrows.

생성이 완료되고 나면 새로 생성된 Notebook Server 를 사용할 수 있는 상태가 되어 오른쪽의 “CONNECT” 메뉴가 활성화됩니다.

This screenshot is similar to the previous one but focuses on the "CONNECT" menu item. The "CONNECT" button for the "torch-cuda-sudo" notebook is now highlighted with a red circle. The rest of the interface is identical to the first screenshot, showing the same notebook details and the "CONNECT" button.

이 “CONNECT” 메뉴를 클릭하면 브라우저의 새로운 탭화면에 다음과 같은 JupyterLab 화면이 나타납니다.



JupyterLab 화면에서는 위 그림과 같이 Jupyter Notebook, Python Console, Linux Terminal, Python File 등과 같은 기능들을 활용하여 인공지능 프로그래밍과 빅데이터 프로그래밍을 수행할 수 있습니다.

## 2) Jupyter Notebook에서 GPU 확인

JupyterLab UI에서 터미널을 실행하면 가상 시스템에 접속하여 Linux 명령어를 실행할 수 있습니다. 터미널에서 nvidia-smi 명령을 실행하면 현재 사용 가능한 GPU 정보를 볼 수 있으며 notebook 파일이나 python 파일에서 이 GPU를 사용할 수 있습니다.

```

File Edit View Run Kernel Git Tabs Settings Help
+ Terminal 1 + 
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

(base) joyyan@torch-test-0:~$ nvidia-smi
Fri May 30 16:57:24 2025
+-----+-----+-----+
| NVIDIA-SMI 550.120 | Driver Version: 550.120 | CUDA Version: 12.4 |
+-----+-----+-----+
| GPU Name Persistence-M Bus-Id Disp.A Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap | Memory-Usage | GPU-Util Compute M. |
| 0 NVIDIA GeForce RTX 3080 Off 00000000:01:00.0 Off | 201MiB / 16384MiB | 0% E. Process N/A |
| N/A 37C P8 9W / 90W |                         |                         |
+-----+-----+-----+
Processes:
+-----+-----+-----+-----+-----+
| GPU GI CI PID Type Process name | GPU Memory Usage |
| ID ID          |                         |
+-----+-----+-----+-----+
(base) joyyan@torch-test-0:~$ 

```

### 3) Jupyter Notebook에서 딥러닝 실행

JupyterLab UI에서 다음과 같이 노트북 파일을 만들고 딥러닝 코드를 만들어 실행할 수 있습니다.

```

File Edit View Run Kernel Git Tabs Settings Help
mnist.ipynb x 1-data-process.ipynb x +
+ × 🔍 Code ⌂ Notebook Python 3 (ipykernel) 
[1]: import torch
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
from torchvision import datasets, transforms
from torch.utils.data import DataLoader

[2]: # Check if CUDA is available and set device
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

[4]: # Hyperparameters
batch_size = 64
learning_rate = 0.01
epochs = 10

[5]: # MNIST dataset
train_dataset = datasets.MNIST(root='./data', train=True, transform=transforms.ToTensor(), download=True)
test_dataset = datasets.MNIST(root='./data', train=False, transform=transforms.ToTensor())

```

Downloading http://yann.lecun.com/exdb/mnist/train-images-idx3-ubyte.gz  
Failed to download (trying next):  
HTTP Error 403: Forbidden

Downloading https://oss-eu-west-1.aliyuncs.com/mnist/train-images-idx3-ubyte.gz  
Downloading https://oss-eu-west-1.aliyuncs.com/mnist/train-images-idx3-ubyte.gz to ./data//MNIST/raw/train-images-idx3-ubyte.gz  
100% [██████████] 9912422 [00:01<00:00, 5644135.34it/s]  
Extracting ./data//MNIST/raw/train-images-idx3-ubyte.gz to ./data//MNIST/raw

Downloading http://yann.lecun.com/exdb/mnist/train-labels-idx1-ubyte.gz  
Failed to download (trying next):  
HTTP Error 403: Forbidden

Downloading https://oss-eu-west-1.aliyuncs.com/mnist/train-labels-idx1-ubyte.gz  
Downloading https://oss-eu-west-1.aliyuncs.com/mnist/train-labels-idx1-ubyte.gz to ./data//MNIST/raw/train-labels-idx1-ubyte.gz  
100% [██████████] 28881/28881 [00:00<00:00, 166664.18it/s]  
Extracting ./data//MNIST/raw/train-labels-idx1-ubyte.gz to ./data//MNIST/raw

Downloading http://yann.lecun.com/exdb/mnist/t10k-images-idx3-ubyte.gz  
Failed to download (trying next):  
HTTP Error 403: Forbidden

### 4) Notebook Server 정지하기

생성한 Notebook Server를 사용하지 않을 때는 정지하는 것이 좋습니다. 정지하지 않으면 기본적으로 시스템 자원을 점유하므로 다른 Notebook Server를 실행할 수 있는 자원이 부족할 수 있습니다. Notebooks 메뉴 화면에서 정지 버튼을 클릭하면 Notebook Server가 정지됩니다.

The screenshot shows the Kubeflow Notebooks interface. At the top, there are navigation icons and the text "Kubeflow" and "npark (Owner)". On the right, there is a "New Notebook" button and a user icon. Below this is a section titled "Notebooks" with a "Filter" input field. A table lists a single notebook entry:

Status	Name ↑	Type	Created at	Last activity	Image	GPUs	CPUs	Memory	CONNECT
✓	torch-cuda-sudo	jupyter	6 days ago	-	jupyter-pytorch-cu...	1	2	8.0 Gi	<span>CONNECT</span>

Below the table are pagination controls: "Items per page: 10", "1 - 1 of 1", and navigation arrows. A red circle highlights the "CONNECT" button, which is currently labeled "CONNECT". A blue callout box with the Korean text "정지/시작 버튼" (Stop/Start button) points to this button.

Notebook Server 가 정지된 상태에서는 정지 버튼이 시작 버튼으로 변경됩니다. 다시 시작하기 위해서는 이 시작 버튼을 클릭합니다.

## 5) Notebook Server 삭제하기

더 이상 사용하지 않을 Notebook Server 를 삭제할 수 있습니다. Notebook Server 목록의 오른쪽에 있는 “휴지통 모양”의 아이콘을 클릭하여 Notebook Server 를 삭제할 수 있습니다.

This screenshot is similar to the previous one, showing the Kubeflow Notebooks interface with the same notebook entry. The "CONNECT" button has been replaced by a trash bin icon, indicating the notebook is no longer running. A red circle highlights this trash bin icon, and a blue callout box with the Korean text "삭제 버튼" (Delete button) points to it.

## 6) Volumes 관리

Notebook Server 가 사용하는 스토리지는 Kubeflow central dashboard 의 “Volumes” 메뉴에서 볼 수 있습니다. Notebook Server 를 삭제한 후에도 스토리지(Volume)은 남아 있으므로 필요 없는 경우 볼륨을 삭제해야 합니다.

The screenshot shows the Kubeflow central dashboard interface. On the left, there is a sidebar with the following navigation options:

- Home
- Notebooks** (selected)
- TensorBoards
- Volumes (highlighted with a red oval)
- Katib Experiments
- KServe Endpoints
- Pipelines
- Manage Contributors
- GitHub
- Documentation

Below the sidebar, the main content area is titled "Notebooks". It displays a table with one row of data:

Status	Name	Type	Created at	Last activity	Image	GPUs	CPUs	Memory	Actions
✓	torch-cuda-su...	notebook	6 days ago	-	jupyter-pytorch-...	1	2	8.0 Gi	<b>CONNECT</b>

At the bottom of the content area, there are pagination controls: "Items per page: 10", "1 - 1 of 1", and navigation arrows.

## 3. Contributor 관리

Kubeflow 는 사용자 별로 하나의 워크스페이스 내에서 필요한 작업을 할 수 있습니다. 협업과 교육을 위해서 자신의 워크스페이스를 다른 사용자에게 공유하는 기능이 제공됩니다. 사용자는 Contributor 관리 기능을 통해 자신의 워크스페이스를 제어할 수 있는 권한을 다른 사용자에게 부여할 수 있습니다. 이 기능을 이용하여 상호 협업을 하거나 타인에게 자신의 작업에 대한 도움을 받을 수 있습니다.

The screenshot shows the Kubeflow interface with a dark blue sidebar on the left containing various navigation links: Home, Notebooks, TensorBoards, Volumes, Katib Experiments, KServe Endpoints, Pipelines, Manage Contributors, GitHub, and Documentation. The 'Manage Contributors' link is highlighted. The main content area has a header 'Manage Contributors'. It includes sections for 'Account Information' (email: npark@starcell.co.kr) and 'Profile Memberships' (listing 'Owner' role in 'npark' namespace). Below this is a section titled 'Contributors for - npark' with a search bar for 'Email Addresses' and a button 'Add by email address'.

사용자는 자신을 Contributor로 등록하여 권한을 부여한 사용자의 워크스페이스에 접근하여 사용할 수 있습니다. 위 부분의 네임스페이스 선택 메뉴에서 자신이 접근가능한 네임스페이스의 목록을 볼 수 있습니다.

The screenshot shows the Kubeflow interface with a dark blue sidebar on the left containing various navigation links: Home, Notebooks, TensorBoards, Volumes, Katib Experiments, KServe Endpoints, Pipelines, Manage Contributors, GitHub, and Documentation. The 'Notebooks' link is highlighted. The main content area has a header 'npark (Owner)' and a search bar for 'Name or value'. It lists notebooks in the 'blue' namespace, grouped under 'npark (Owner)'. The table columns include Type, Created at, Last activity, Image, GPUs, CPUs, Memory, and actions (CONNECT, RESTART, STOP, DELETE). The entries are:

Type	Created at	Last activity	Image	Gpus	Cpus	Memory	Action
r-test	2 months ago	- rstudio-tidyver...	0	1	2.0 Gi	CONNECT	▶ 🔍
torch-cuda-s...	10 months ago	- jupyter-ptorch...	1	2	8.0 Gi	CONNECT	▣ 🔍
vs-code-test	8 months ago	- codeserver-pt...	0	1	2.0 Gi	CONNECT	▶ 🔍

# III. 트러블 슈팅

Notebook Server 가 생성되지 않을 때

자신의 계정이 사용할 수 있는 자원이 있는지 확인합니다. 자신에게 할당된 자원이 부족한 경우 Notebook Server 생성이 안되고 오류가 발생합니다. 자신에게 할당된 자원에 대해 모르는 경우 관리자에게 문의합니다.

JupyterLab 의 터미널에서 nvidia-smi 명령을 실행할 때 다음과 같은 오류가 날 경우

Failed to initialize NVML: Unknown Error

Notebook Server 를 정지했다가 다시 시작합니다.