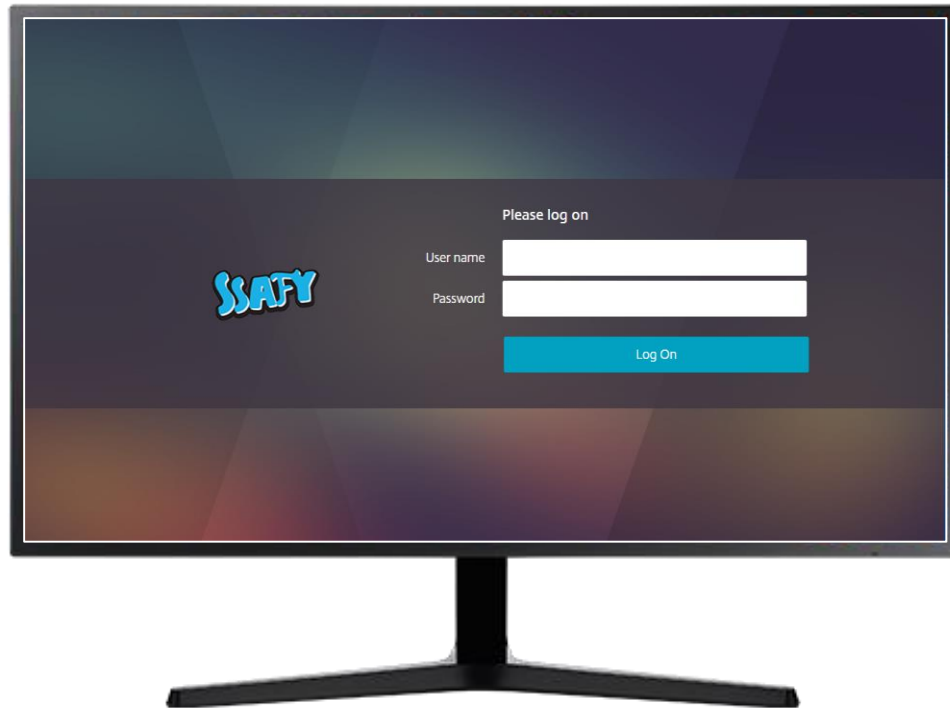




# GPU 개발 서버 사용 안내

2022.09.05

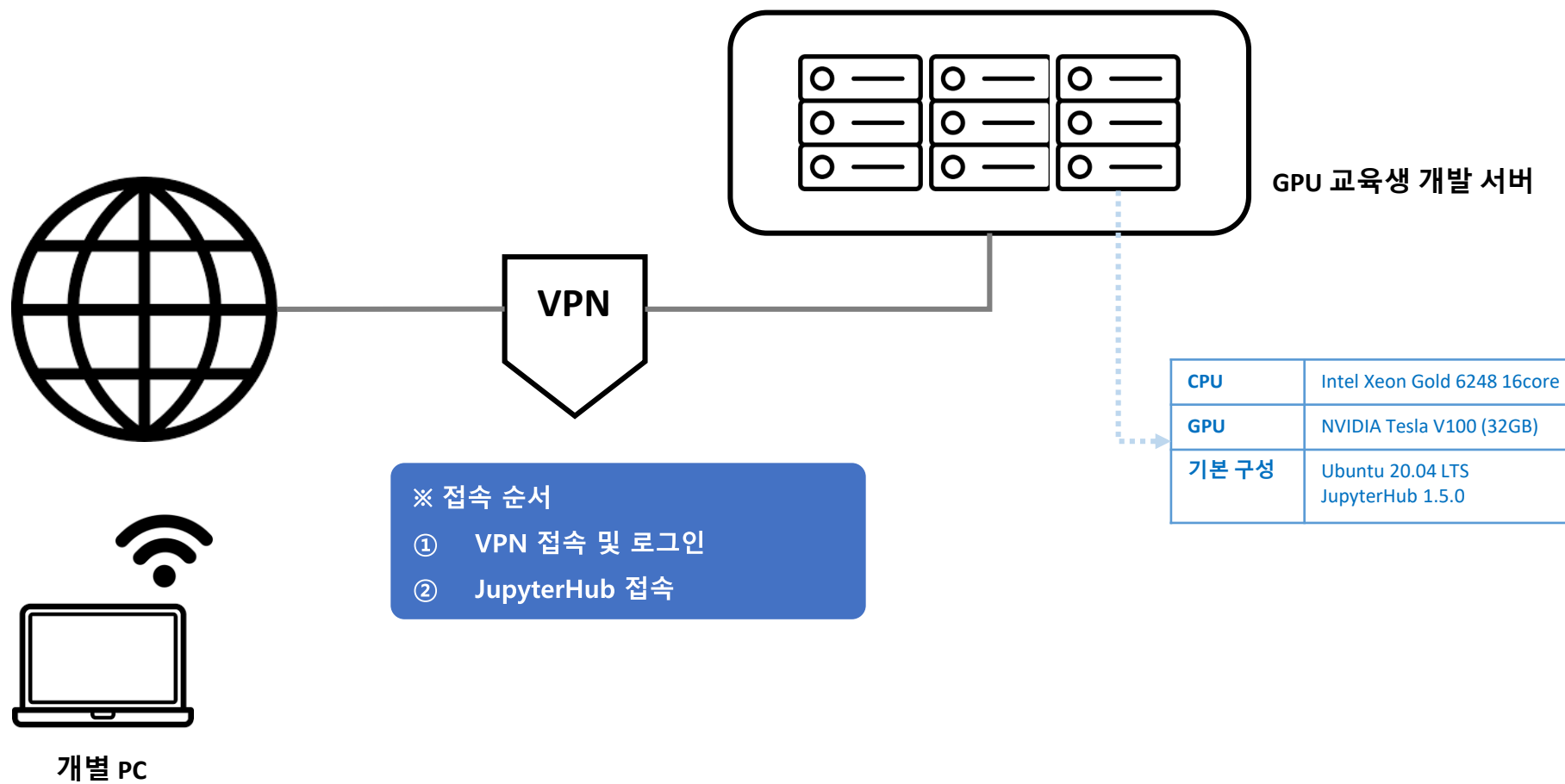


# Contents

- I. 소개
- II. 접속 방법
- III. 환경 설정
- IV. 주의사항

# I. 소개

교육생 개발 서버: 인공지능 학습을 위해 제공되는 일반 PC 대비 고사양의 GPU 장착 서버



## II. 접속 방법

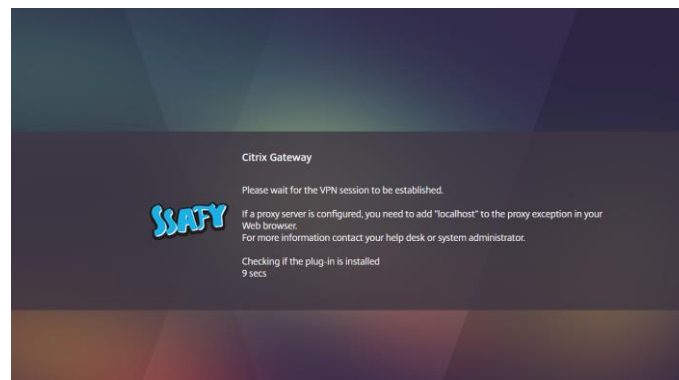
### 1. VPN 접속

#### ① 웹 브라우저를 열어 VPN 서버로 접속

<https://server.ssafy.com>

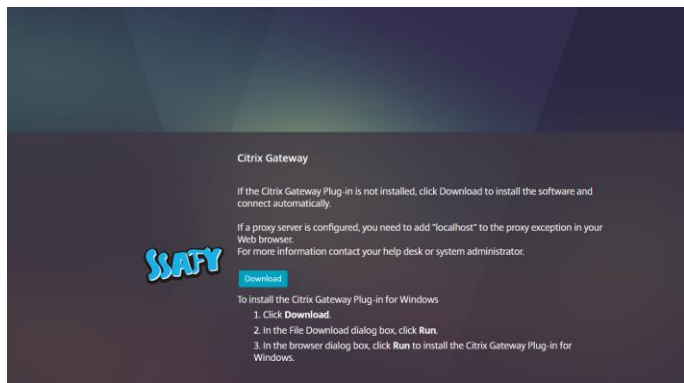


#### ② VPN 플러그인 설치 여부를 자동으로 체크

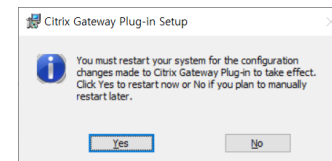


#### i. 플러그인이 설치되어 있지 않은 경우

##### ① 플러그인 설치 파일 다운로드



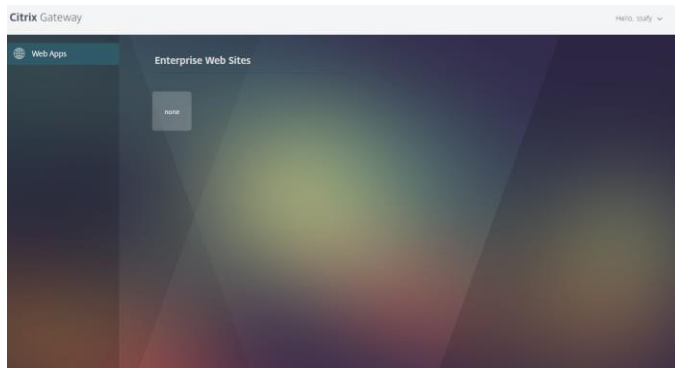
##### ② 설치 진행 및 및 재부팅



## II. 접속 방법

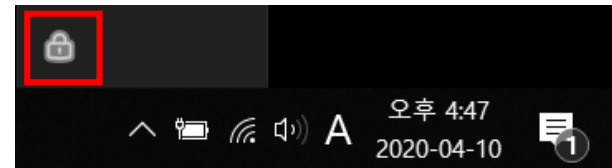
### ii. 플러그인이 설치되어 있는 경우

#### ① VPN 로그인 완료 화면 확인

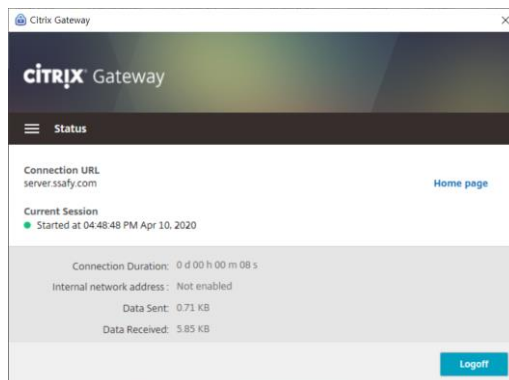


#### ② VPN 로그인 완료 시

작업 표시줄에서 플러그인 아이콘 확인 가능



#### ③ 플러그인 아이콘을 더블 클릭하여 Current Session으로 접속 상태 확인



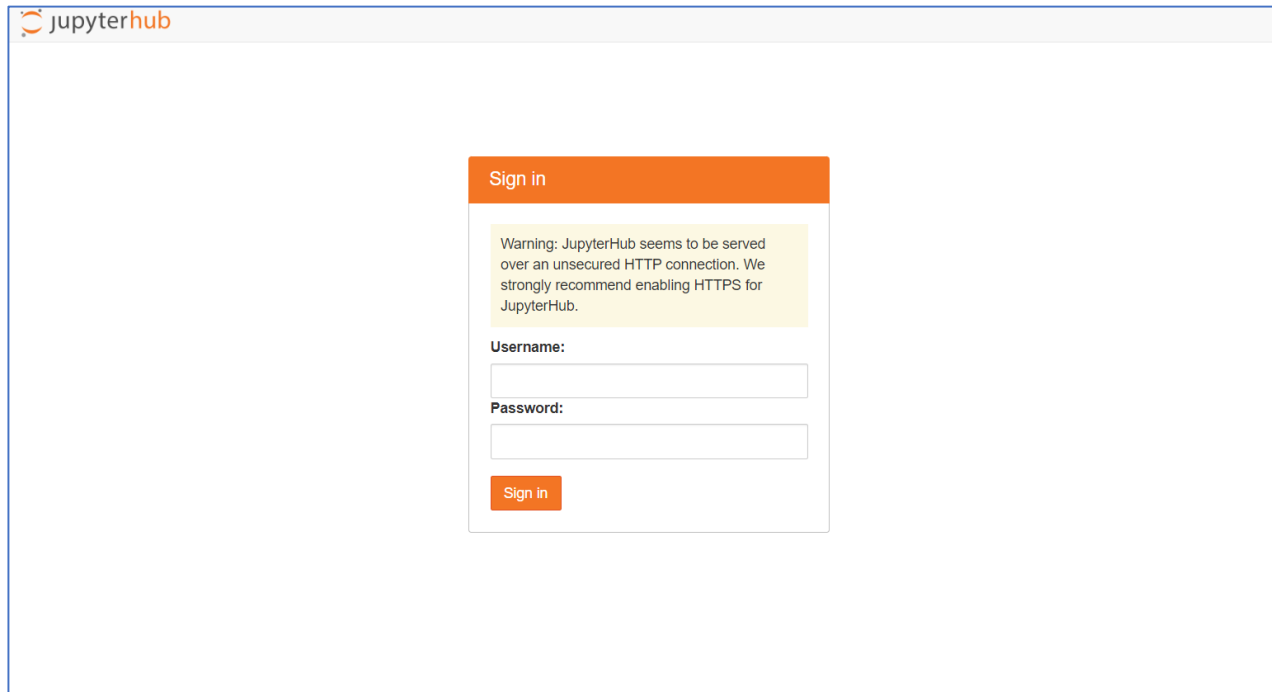
## II. 접속 방법

### 2. Jupyter Hub 접속

(VPN 연결된 상태에서)

① 웹 브라우저를 열어 Jupyter Hub로 접속

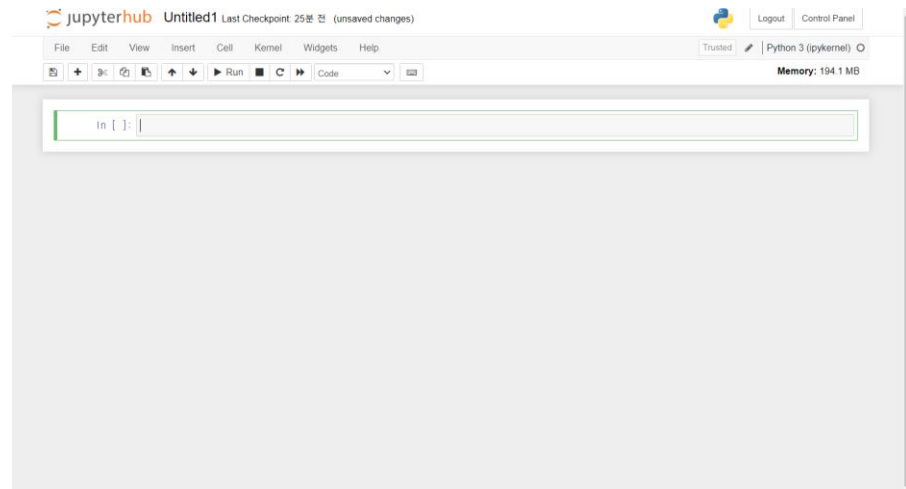
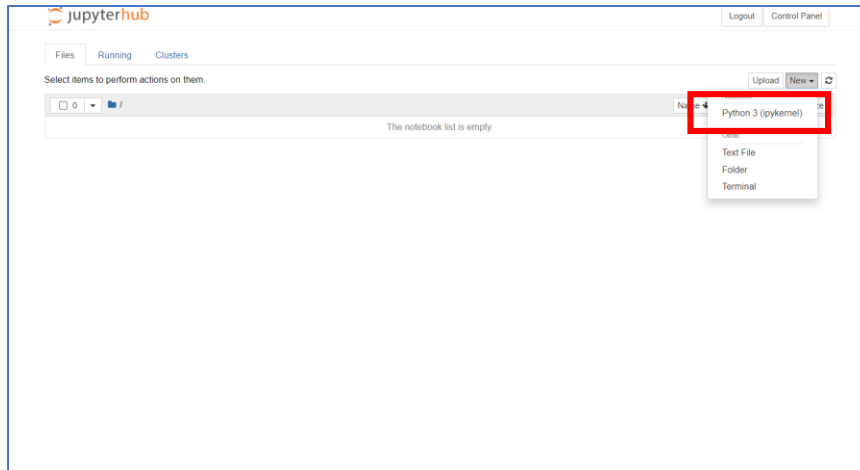
- 접속 IP: 별도 안내
- Username: 별도 안내
- Password: 별도 안내



The screenshot shows the JupyterHub web interface. At the top left, the 'jupyterhub' logo is visible. In the center, there is a 'Sign in' form. The form has an orange header with the text 'Sign in'. Below the header, there is a yellow warning box that reads: 'Warning: JupyterHub seems to be served over an unsecured HTTP connection. We strongly recommend enabling HTTPS for JupyterHub.' Under the warning box, there are two input fields: 'Username:' and 'Password:'. At the bottom of the form, there is an orange 'Sign in' button.

## II. 접속 방법

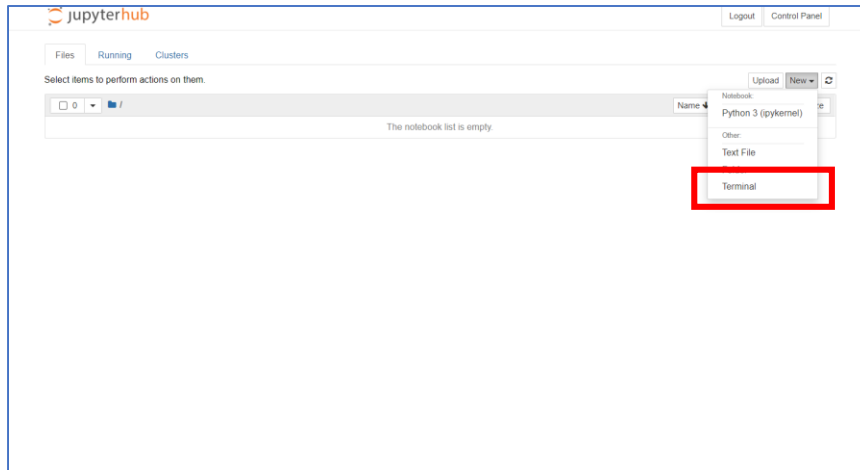
### ② Jupyter Notebook 파일 만들기



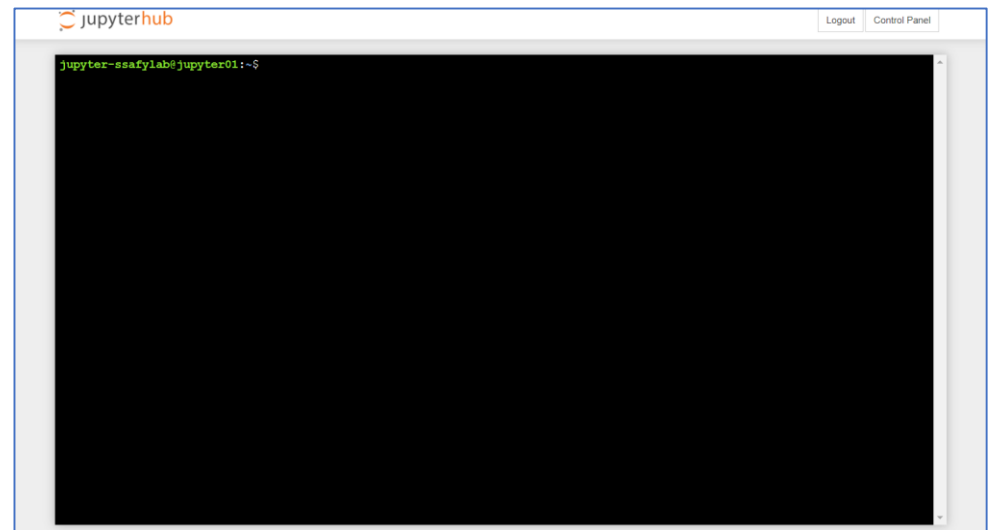
※ 일반적인 Jupyter Notebook 사용 동일

## II. 접속 방법

### ③ Terminal 창 열기



※ python 버전 변경, conda 설치 및  
ipykernel 설정 등을 user 권한 내에서  
자유롭게 사용 가능





### III. 환경 설정

#### 1. 서버 기본 정보 확인

##### ① home 디렉토리 확인

```
team1@gpu-server01:~$ echo $HOME
/home/team1
```

##### ② 하드웨어 정보 확인

```
team1@gpu-server01:~$ lshw -short | egrep 'processor|display'
/0/1          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/2          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/3          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/4          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/5          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/6          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/7          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/8          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/9          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/a          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/b          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/c          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/d          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/e          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/f          processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/10         processor Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz
/0/100/f      display   SVGA II Adapter
/0/100/15/0   display   GV100GL [Tesla V100 PCIe 32GB]
```

### III. 환경 설정

#### ③ GPU 정보 확인

```
(base) team1@gpu-server01:~$ nvidia-smi
```

NVIDIA-SMI 460.56 Driver Version: 460.56 CUDA Version: 11.2									
GPU	Name	Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr. ECC			
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage	GPU-Util	Compute M.			
						MIG M.			
0	Tesla V100-PCIE...	Off	00000000:03:00.0	Off	0				
N/A	29C	P0	23W / 250W	4MiB / 32510MiB	0%	Default			
						N/A			

Processes:						
GPU	GI	CI	PID	Type	Process name	GPU Memory Usage
	ID	ID				
0	N/A	N/A	1512	G	/usr/lib/xorg/Xorg	4MiB

#### ④ CUDA toolkit 확인

```
(base) team1@gpu-server01:~$ nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2021 NVIDIA Corporation
Built on Sun_Feb_14_21:12:58_PST_2021
Cuda compilation tools, release 11.2, V11.2.152
Build cuda_11.2.r11.2/compiler.29618528_0
```

## IV. 환경 설정

### 2. 프로젝트 학습 환경 설정

#### ① Jupyter Notebook Kernel 추가

```
team1@gpu-server01:~$ jupyter kernelspec list
Available kernels:
  python3      /opt/tljh/user/share/jupyter/kernels/python3

team1@gpu-server01:~$ pip install ipykernel

team1@gpu-server01:~$ python -m ipykernel install --user --name 가상 환경명 --display-name 출력명
```

★ conda 가상환경을 ipykernel에 추가할 경우 Anaconda 설치 및 activate 상태에서 위 커맨드 수행

#### ② 필요한 패키지, 라이브러리 등은 User 권한으로 홈 디렉토리에 다운로드 또는 설치하여 사용

## IV. 환경 설정

### 3. 프로젝트 학습 환경 설정

#### ① Anaconda 가상 환경 생성

```
(base) team1@gpu-server01:~$ conda --version
conda 4.9.2
(base) team1@gpu-server01:~$ conda create -n 가상_환경명
(base) team1@gpu-server01:~$ conda activate 가상_환경명
(가상_환경명) team1@gpu_server01:~$
```

★ conda가 설치되어 있지 않거나 다른 버전을 원할 경우 직접 설치하여 사용해도 무방합니다.

#### ② 필요한 패키지, 라이브러리 등은 Anaconda 가상 환경 내에 설치하여 사용

★ 서버의 sudo 권한이 반드시 필요한 경우는 SSAFY GIT Q&A 게시판에 필요 사유를 반드시 포함하여 요청

