

# AI KU A100 การใช้งาน Conda

## 1. การใช้งาน Python ด้วย Conda

- 1.1 คำสั่งสำหรับเริ่มต้นการใช้งาน Conda
- 1.2 การสร้าง Virtual Environment ด้วย Conda
- 1.3 การใช้งาน Conda Virtual Environment
- 1.4 การติดตั้ง Library ภายใน Conda Virtual Environment
- 1.5 การสร้าง Conda Environment เพื่อใช้งานบน Jupyter Notebook
- 1.6 การลบ Virtual Environment ด้วย Conda
- 1.7 การเลิกใช้งาน Conda Virtual Environment

## 2. การรันงาน Jupyter Notebook job ผ่านโปรแกรมจัดการทรัพยากร SLURM Workload Manager และเข้าใช้งานผ่าน SSH Tunnel

- 2.1 การรันงาน Jupyter Notebook job ผ่านโปรแกรมจัดการทรัพยากร SLURM Workload Manager
  - 2.1.1) Start Jupyter Notebook Batch Job ผ่าน Slurm
  - 2.1.2) Start SSH Tunnel
    - การทำ SSH Tunnel บน MacOC และ Linux
    - การทำ SSH Tunnel บน Windows
  - 2.1.3) เข้าใช้งานผ่าน Browser บนเครื่องของผู้ใช้งาน
- 2.2 การหยุดใช้งาน Jupyter Notebook

## 1. การใช้งาน Python ด้วย Conda

### 1.1 คำสั่งสำหรับเริ่มต้นการใช้งาน Conda

1. รันคำสั่ง `module load anaconda3`
2. รันคำสั่ง `conda init --all`
3. Logout และ Login เข้ามาใหม่

```
jsorawid@br1:~$ module load anaconda3/
jsorawid@br1:~$ conda init --all
no change      /cm/shared/apps/anaconda3/condabin/conda
no change      /cm/shared/apps/anaconda3/bin/conda
no change      /cm/shared/apps/anaconda3/bin/conda-env
no change      /cm/shared/apps/anaconda3/bin/activate
no change      /cm/shared/apps/anaconda3/bin/deactivate
no change      /cm/shared/apps/anaconda3/etc/profile.d/cond
```

```

a.sh
no change      /cm/shared/apps/anaconda3/etc/fish/conf.d/conda.fish
no change      /cm/shared/apps/anaconda3/shell/condabin/Conda.psm1
no change      /cm/shared/apps/anaconda3/shell/condabin/conda-hook.ps1
no change      /cm/shared/apps/anaconda3/lib/python3.11/site-packages/xontrib/conda.xsh
no change      /cm/shared/apps/anaconda3/etc/profile.d/conda.csh
modified       /home/jsorawid/.bashrc
modified       /home/jsorawid/.zshrc
modified       /home/jsorawid/.config/fish/config.fish
modified       /home/jsorawid/.xonshrc
modified       /home/jsorawid/.tcshrc

```

==> For changes to take effect, close and re-open your current shell. <==

```

jsorawid@br1:~$
logout
root@br1:~# su - jsorawid
(base) jsorawid@br1:~$

```

## 1.2 การสร้าง Virtual Environment ด้วย Conda

สร้าง Virtual Environment ใน Conda ด้วยคำสั่ง `conda create -n <name>`

```

(base) jsorawid@br1:~$ conda create -n python3.6 python=3.6
.
.
.

```

The following **NEW** packages will be **INSTALLED**:

```

    _libgcc_mutex          pkgs/main/linux-64::_libgcc_mutex-0.1-
main

```

```

_openmp_mutex      pkgs/main/linux-64::_openmp_mutex-5.1-
1_gnu
  ca-certificates  pkgs/main/linux-64::ca-certificates-20
24.3.11-h06a4308_0
  certifi          pkgs/main/linux-64::certifi-2021.5.30-
py36h06a4308_0
  ld_impl_linux-64 pkgs/main/linux-64::ld_impl_linux-64-
2.38-h1181459_1
  libffi          pkgs/main/linux-64::libffi-3.3-he6710b
0_2
  libgcc-ng       pkgs/main/linux-64::libgcc-ng-11.2.0-h
1234567_1
  libgomp         pkgs/main/linux-64::libgomp-11.2.0-h12
34567_1
  libstdcxx-ng    pkgs/main/linux-64::libstdcxx-ng-11.2.
0-h1234567_1
  ncurses         pkgs/main/linux-64::ncurses-6.4-h6a678
d5_0
  openssl        pkgs/main/linux-64::openssl-1.1.1w-h7f
8727e_0
  pip             pkgs/main/linux-64::pip-21.2.2-py36h06
a4308_0
  python         pkgs/main/linux-64::python-3.6.13-h12d
ebd9_1
  readline       pkgs/main/linux-64::readline-8.2-h5eee
18b_0
  setuptools     pkgs/main/linux-64::setuptools-58.0.4-
py36h06a4308_0
  sqlite         pkgs/main/linux-64::sqlite-3.45.3-h5ee
e18b_0
  tk             pkgs/main/linux-64::tk-8.6.14-h39e8969
_0
  wheel          pkgs/main/noarch::wheel-0.37.1-pyhd3eb
1b0_0
  xz             pkgs/main/linux-64::xz-5.4.6-h5eee18b_
1
  zlib           pkgs/main/linux-64::zlib-1.2.13-h5eee1
8b_1

```

```
Proceed ([y]/n)? y
```

## 1.3 การใช้งาน Conda Virtual Environment

ใช้คำสั่ง `conda activate <package_name>` เมื่อเรียกใช้งานสำเร็จ จะปรากฏชื่อ Environment นั้น ๆ ในวงเล็บอยู่ที่ซ้ายสุดของบรรทัด เช่น (base) หรือ (python3.6)

```
(base) jsorawid@br1:~$ conda env list
# conda environments:
#
base                *  /cm/shared/apps/anaconda3
DeepBSA             /cm/shared/apps/anaconda3/envs/Dee
pBSA
MaSuRCA             /cm/shared/apps/anaconda3/envs/MaS
uRCA
QIIME2              /cm/shared/apps/anaconda3/envs/QII
ME2
QUAST               /cm/shared/apps/anaconda3/envs/QUA
ST
bioapps             /cm/shared/apps/anaconda3/envs/bio
apps
busco               /cm/shared/apps/anaconda3/envs/bus
co
cutadapt            /cm/shared/apps/anaconda3/envs/cut
adapt
deepvariant         /cm/shared/apps/anaconda3/envs/dee
pvariant
pankmer             /cm/shared/apps/anaconda3/envs/pan
kmer
python2.7           /cm/shared/apps/anaconda3/envs/pyt
hon2.7
python3.8           /cm/shared/apps/anaconda3/envs/pyt
hon3.8

(base) jsorawid@br1:~$ conda activate bioapps
```

```
(bioapps) jsorawid@br1:~$ python -V
Python 3.12.2
```

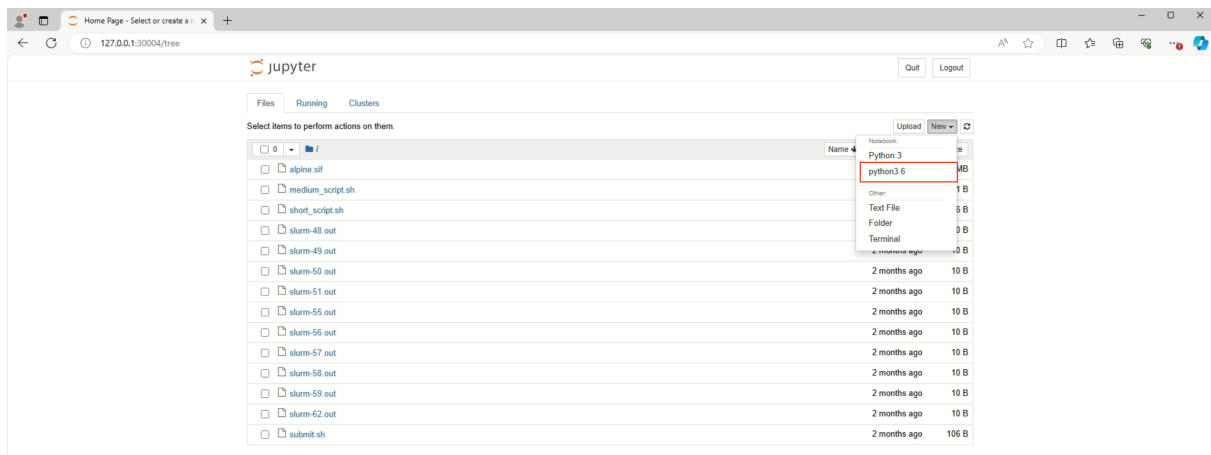
## 1.4 การติดตั้ง Library ภายใน Conda Virtual Environment

ใช้คำสั่ง `conda install <package_name>`

```
(python3.6) (base) jsorawid@br1:~$ conda install pandas
```

## 1.5 การสร้าง Conda Environment เพื่อใช้งานบน Jupyter Notebook

1. เรียกใช้ Environment ที่ต้องการ `conda activate python3.6`
2. ติดตั้ง Jupyter Notebook `conda install notebook`
3. ติดตั้ง ipykernel package `conda install ipykernel`
4. เรียกคำสั่ง `python -m ipykernel install --user --name=<any_name_for_kernel>`
5. เปิด Jupyter Notebook และทำ SSH tunnel [ตามคู่มือในส่วนถัดไป](#)
6. ตรวจสอบ Kernel บน Jupyter Notebook



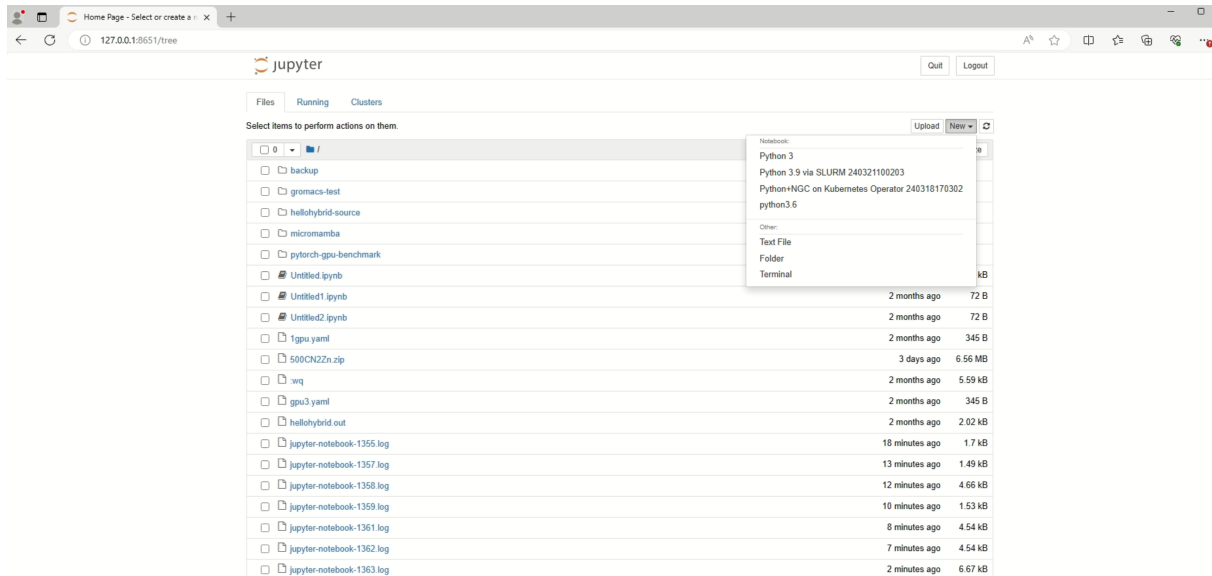
## 1.6 การลบ Virtual Environment ด้วย Conda

ใช้คำสั่ง `conda env remove --name <environment_name>`

## 1.7 การเลิกใช้งาน Conda Virtual Environment

ใช้คำสั่ง `conda deactivate` เมื่อไม่ต้องการใช้งาน

## 2. การรันงาน Jupyter Notebook job ผ่านโปรแกรมจัดการทรัพยากร SLURM Workload Manager และเข้าใช้งานผ่าน SSH Tunnel



ผู้ใช้งานสามารถรันงาน Jupyter Notebook บนเครื่อง compute และเข้าใช้งานผ่านเครื่องของตนเองได้ โดย Slurm จะทำหน้าที่จัดสรรทรัพยากรให้กับงานเหล่านี้

ในเบื้องต้นเราได้ติดตั้ง Jupyter Notebook ไว้ใน Conda environment เรียบร้อยแล้ว

### 2.1 การรันงาน Jupyter Notebook job ผ่านโปรแกรมจัดการทรัพยากร SLURM Workload Manager

ประกอบด้วย 3 ขั้นตอน

2.1.1) Start Jupyter Notebook Batch Job ผ่าน Slurm

2.1.2) Start SSH Tunnel

2.1.3) เข้าใช้งานผ่าน Browser บนเครื่องของผู้ใช้งาน

#### 2.1.1) Start Jupyter Notebook Batch Job ผ่าน Slurm

ด้านล่างเป็นตัวอย่างสคริปต์สำหรับ Submit งานของ Jupyter Notebook (ตัวอย่างไฟล์วางอยู่ที่ `/cm/shared/examples/notebook.sh`) เป็น Batch Job ผู้ใช้งานควรปรับค่าต่าง ๆ ของ Slurm (`#SBATCH`) ตามความเหมาะสม เมื่อแก้ไขไฟล์สคริปต์เสร็จแล้ว ให้ Submit งานด้วยคำสั่ง

```
sbatch /cm/shared/examples/notebook.sh
```

```

root@br1:~# cat /cm/shared/examples/notebook.sh
#!/bin/bash
#SBATCH --partition gpuq
#SBATCH --gres=gpu:1
#SBATCH --mem-per-cpu 8G
#SBATCH --time 1-0:00:00
#SBATCH --job-name jupyter-notebook
#SBATCH --output jupyter-notebook-%J.log

# get tunneling info
XDG_RUNTIME_DIR=""
port=$(shuf -i8000-9999 -n1)
node=$(hostname -s)
user=$(whoami)
cluster="br1.paas.ku.ac.th"
#cluster=$(hostname -f | awk -F"." '{print $2}')

# print tunneling instructions jupyter-log
echo -e "

MacOS or linux terminal command to create your ssh tunnel
ssh -N -L ${port}:${node}:${port} ${user}@${cluster}

Windows MobaXterm info
Forwarded port:same as remote port
Remote server: ${node}
Remote port: ${port}
SSH server: ${cluster}
SSH login: $user
SSH port: 22

Use a Browser on your local machine to go to:
localhost:${port} (prefix w/ https:// if using password)
"

# You have to initialize anaconda3 first before using conda
a.
# The following two lines used to activate your conda envir

```

```
onment i.e. my env is called 'python3.6'  
module list  
source activate python3.6  
  
# DON'T USE ADDRESS BELOW.  
# DO USE TOKEN BELOW  
jupyter-notebook --no-browser --port=${port} --ip=${node}
```

```
(base) [peeranon@slurmmaster ~]$ sbatch /cm/shared/example  
s/notebook.sh  
Submitted batch job 1367
```

## 2.1.2) Start SSH Tunnel

เมื่อ Submit งานเรียบร้อยแล้ว Jupyter Notebook Server จะเริ่มทำงานและพร้อมแก่การเชื่อมต่อ โดยผู้ใช้งานสามารถรันคำสั่ง `squeue -u $USER` เพื่อตรวจสอบงานที่ผู้ใช้งานรันอยู่ สถานะที่ขึ้นว่า R ในคอลัมน์ ST บ่งบอกว่างานกำลังรันอยู่ ถ้าขึ้นสถานะ PD ให้รอสักครู่ แล้วจึงจะสามารถเข้าใช้งานได้ ไฟล์ Log ที่จะอธิบายข้อมูลการเข้าใช้งาน จะเป็นชื่อ `jupyter-notebook-[jobid].log` ซึ่ง jobid เป็นเลข Job ของผู้ใช้งานใน Slurm

```
(base) jsorawid@br1:~$ squeue -u jsorawid  
          JOBID PARTITION     NAME     USER ST       TIME  N  
ODES NODELIST(REASON)  
          1367      gpuq jupyter- jsorawid  R       1:25  
1 dgx-01
```

ด้านล่างเป็นตัวอย่างไฟล์ log ที่อธิบายวิธีเข้าใช้งาน



```
(base) jsorawid@br1:~$ cat jupyter-notebook-1367.log
=====
SLURM_JOB_ID = 1367
SLURM_NODELIST = dgx-01
=====

MacOS or linux terminal command to create your ssh tunnel
ssh -N -L 8179:dgx-01:8179 jsorawid@br1.paas.ku.ac.th

Windows MobaXterm info
Forwarded port:same as remote port
Remote server: dgx-01
Remote port: 8179
SSH server: br1.paas.ku.ac.th
SSH login: jsorawid
SSH port: 22

Use a Browser on your local machine to go to:
localhost:8179 (prefix w/ https:// if using password)

Currently Loaded Modulefiles:
 1) gcc/13.1.0  2) slurm/ai-ku-slurm/23.02.6
[I 17:55:03.937 NotebookApp] Serving notebooks from local directory: /home/jsorawid
[I 17:55:03.938 NotebookApp] Jupyter Notebook 6.4.3 is running at:
[I 17:55:03.938 NotebookApp] http://dgx-01:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
[I 17:55:03.938 NotebookApp] or http://127.0.0.1:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
[I 17:55:03.938 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 17:55:03.957 NotebookApp]

To access the notebook, open this file in a browser:
file:///home/jsorawid/.local/share/jupyter/runtime/nbserver-3347576-open.html
Or copy and paste one of these URLs:
http://dgx-01:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
or http://127.0.0.1:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
(base) jsorawid@br1:~$
```

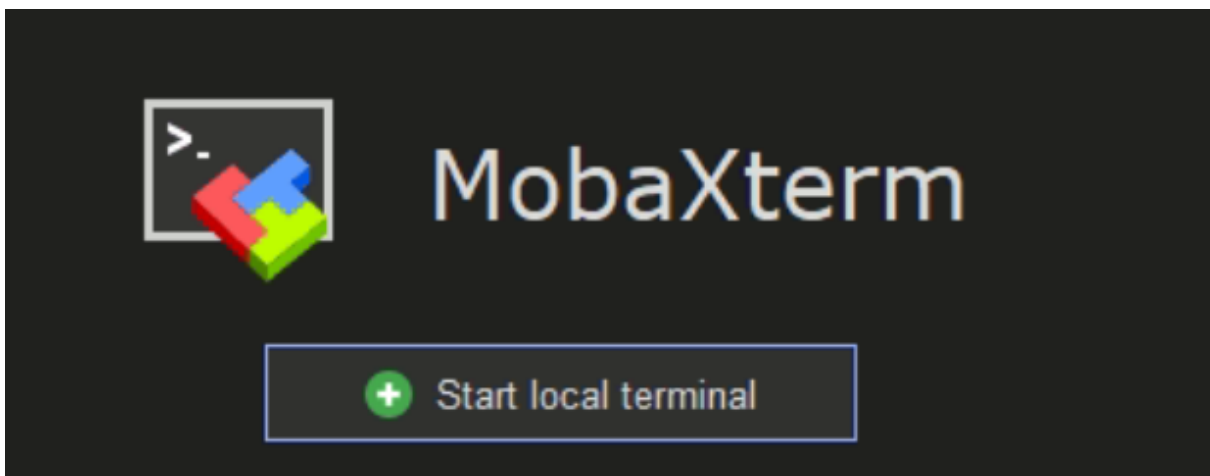
## การทำให้ SSH Tunnel บน MacOC และ Linux

สำหรับผู้ที่ใช้ Mac OS และ Linux ให้เปิด Terminal แล้วรับคำสั่งตามเส้นสีแดง โดยให้ copy คำสั่ง แล้วไปรันบนโปรแกรม Terminal ได้เลย เมื่อรับคำสั่งนี้แล้ว ให้กรอก password ของผู้ใช้งานลงไปให้ถูกต้อง เมื่อกรอกได้ถูกต้อง จะได้ผลตามภาพด้านล่าง

```
08/05/2024 17:57.00 /home/mobaxterm ssh -N -L 8179:dgx-01:8179 jsorawid@br1.paas.ku.ac.th
jsorawid@br1.paas.ku.ac.th's password:
```

## การทำให้ SSH Tunnel บน Windows

ส่วนถ้าใครที่ใช้ Windows ทางเราแนะนำให้ใช้โปรแกรม MobaXterm โดยตรวจสอบข้อมูลเพิ่มเติมเกี่ยวกับการทำให้ SSH Tunnel ได้ โดยทำดังนี้



1. กดเปิด Start local Terminal
2. รับคำสั่งตามเส้นสีแดง เช่น

```
ssh -N -L 8179:dgx-01:8179 jsorawid@br1.paas.ku.ac.th
```

3. ให้กรอก password ของผู้ใช้งานลงไปให้ถูกต้อง เมื่อกรอกได้ถูกต้อง จะได้ผลตามภาพด้านล่าง

```
08/05/2024 17:57.00 /home/mobaxterm ssh -N -L 8179:dgx-01:8179 jsorawid@br1.paas.ku.ac.th
jsorawid@br1.paas.ku.ac.th's password:
```

### 2.1.3) เข้าใช้งานผ่าน Browser บนเครื่องของผู้ใช้งาน

หลังจากที่เปิดใช้งาน SSH Tunnel เสร็จเรียบร้อยแล้ว ในส่วนท้ายของข้อความในไฟล์ log จะมี URL รวมทั้ง token อยู่ ให้ทำการ copy ทั้ง URL ดังภาพในกรอบสีฟ้า ไปวางใน Browser ของตนเอง

```
http://127.0.0.1:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
```

```
(base) jsorawid@br1:~$ cat jupyter-notebook-1367.log
=====
SLURM_JOB_ID = 1367
SLURM_NODELIST = dgx-01
=====

MacOS or linux terminal command to create your ssh tunnel
ssh -N -L 8179:dgx-01:8179 jsorawid@br1.paas.ku.ac.th

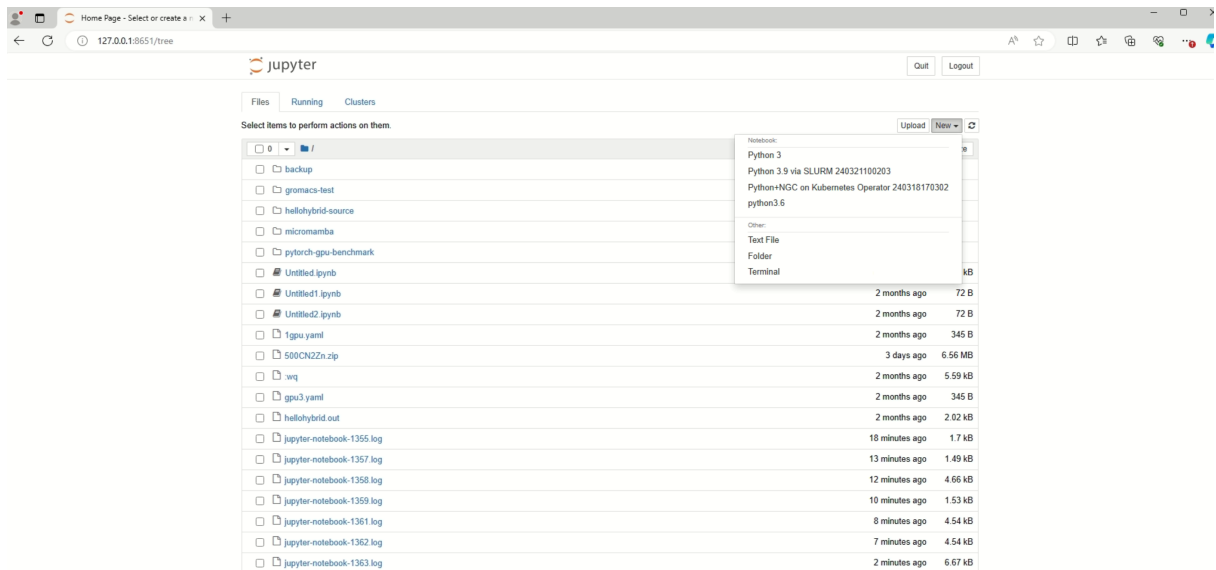
Windows MobaXterm info
Forwarded port:same as remote port
Remote server: dgx-01
Remote port: 8179
SSH server: br1.paas.ku.ac.th
SSH login: jsorawid
SSH port: 22

Use a Browser on your local machine to go to:
localhost:8179 (prefix w/ https:// if using password)

Currently Loaded Modulefiles:
 1) gcc/13.1.0  2) slurm/ai-ku-slurm/23.02.6
[I 17:55:03.937 NotebookApp] Serving notebooks from local directory: /home/jsorawid
[I 17:55:03.938 NotebookApp] Jupyter Notebook 6.4.3 is running at:
[I 17:55:03.938 NotebookApp] http://dgx-01:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
[I 17:55:03.938 NotebookApp] or http://127.0.0.1:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
[I 17:55:03.938 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 17:55:03.957 NotebookApp]

To access the notebook, open this file in a browser:
file:///home/jsorawid/.local/share/jupyter/runtime/nbserver-3347576-open.html
Or copy and paste one of these URLs:
http://dgx-01:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
or http://127.0.0.1:8179/?token=01ab533829b59d81bccd213fec2dad518252767fab558ace
(base) jsorawid@br1:~$
```

ผลของการเข้าใช้งาน เป็นดังภาพด้านล่าง



## 2.2 การหยุดใช้งาน Jupyter Notebook

ในการหยุดใช้งาน Jupyter Notebook ให้ผู้ใช้งานทำการปิด Web Browser และยกเลิกการใช้งาน SSH Tunnel จากนั้นทำการยกเลิกงานดังกล่าวบน Slurm ด้วยคำสั่ง `scancel -f <jobid>`

```
(base) jsorawid@br1:~$ queue -u jsorawid
          JOBID PARTITION    NAME    USER ST      TIME  N
ODES NODELIST(REASON)
          1367      gpuq jupyter- jsorawid  R      1:25
1 dgx-01
(base) jsorawid@br1:~$ scancel -f 1367
(base) jsorawid@br1:~$ queue -u jsorawid
          JOBID PARTITION    NAME    USER ST      TIME  N
ODES NODELIST(REASON)
(base) jsorawid@br1:~$
```