

pipenv system
with or without GPU

If you want use NVIDIA-GPU under pipenv system (1).

- Download and install “Microsoft Visual Studio 2015 Community”
 - <https://visualstudio.microsoft.com/vs/older-downloads/>
 - Necessary join to “Visual Studio Dev Essentials” with account and profile.
 - Select the newest update as “with Update XXX.”

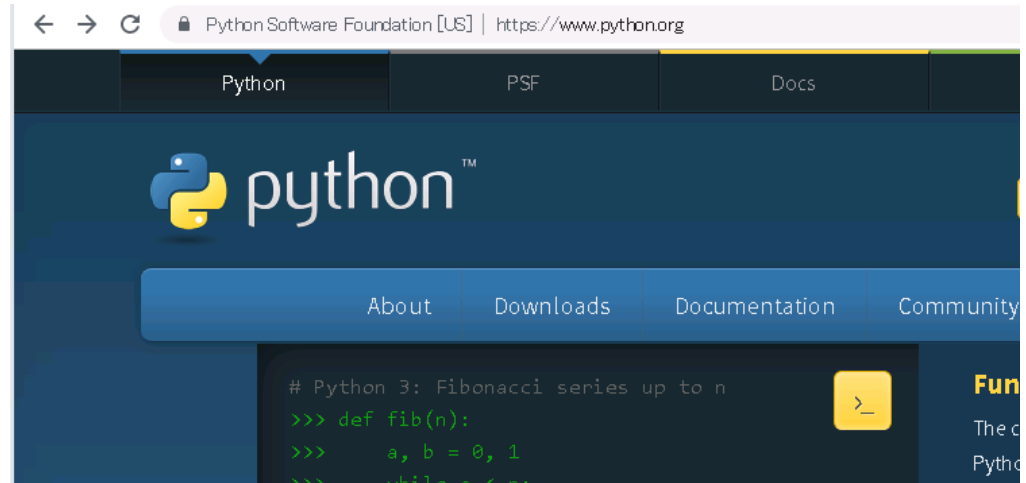
If you want use NVIDIA-GPU under pipenv system (2).

- Check your NVIDIA Driver version.
 - <https://www.nvidia.co.jp/Download/index.aspx>
 - In Japanese
 - <https://www.nvidia.co.jp/Download/index.aspx?lang=jp>
- Check the CUDA Toolkit version for your system.
 - <https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>
- Check the cuDNN, Python and TensorFlow-GPU version.
- Download and install the CUDA toolkit and cuDNN.

Python	CUDA	cuDNN	TensorFlow-GPU
3.6	9	7.31	1.12.0
3.6	8	6	1.4.0

Python3

- Homepage of python
 - <https://www.python.org/>
- Download python3
 - You can use python2, but it's older and depends on the old syntax.



Python

PSF

Docs

PyPI

Jobs

Community



Donate



Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

About

Downloads

Documentation

All releases

Source code

Windows

Mac OSX

Other Platforms

License

Alternative Implementations

Functions Defined

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. [More about defining functions in Python 3](#)

1

2

3

4

5

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

Contribute to the PSF by Purchasing a PyCharm License. All proceeds benefit the PSF.

Donate Now

Get Started

Whether you're new to programming or an experienced developer, it's easy to learn and use Python.

Download

Python source code and installers are available for download for all versions!

Docs

Documentation for Python's standard library, along with tutorials and guides, are available online.

Jobs

Looking for work or have a Python related position that you're trying to hire for? Our **relaunched**

Python Releases for Windows

- [Latest Python 3 Release - Python 3.7.3](#)
- [Latest Python 2 Release - Python 2.7.16](#)

Stable Releases

- [Python 3.7.3 - March 25, 2019](#)

Note that Python 3.7.3 cannot be used on Windows XP or earlier.

- [Python 3.6.8 - Dec. 24, 2018](#)

Note that Python 3.6.8 cannot be used on Windows XP or earlier.

- Download [Windows help file](#)
- Download [Windows x86-64 embeddable zip file](#)
- Download [Windows x86-64 executable installer](#)
- Download [Windows x86-64 web-based installer](#)
- Download [Windows x86 embeddable zip file](#)
- Download [Windows x86 executable installer](#)
- Download [Windows x86 web-based installer](#)

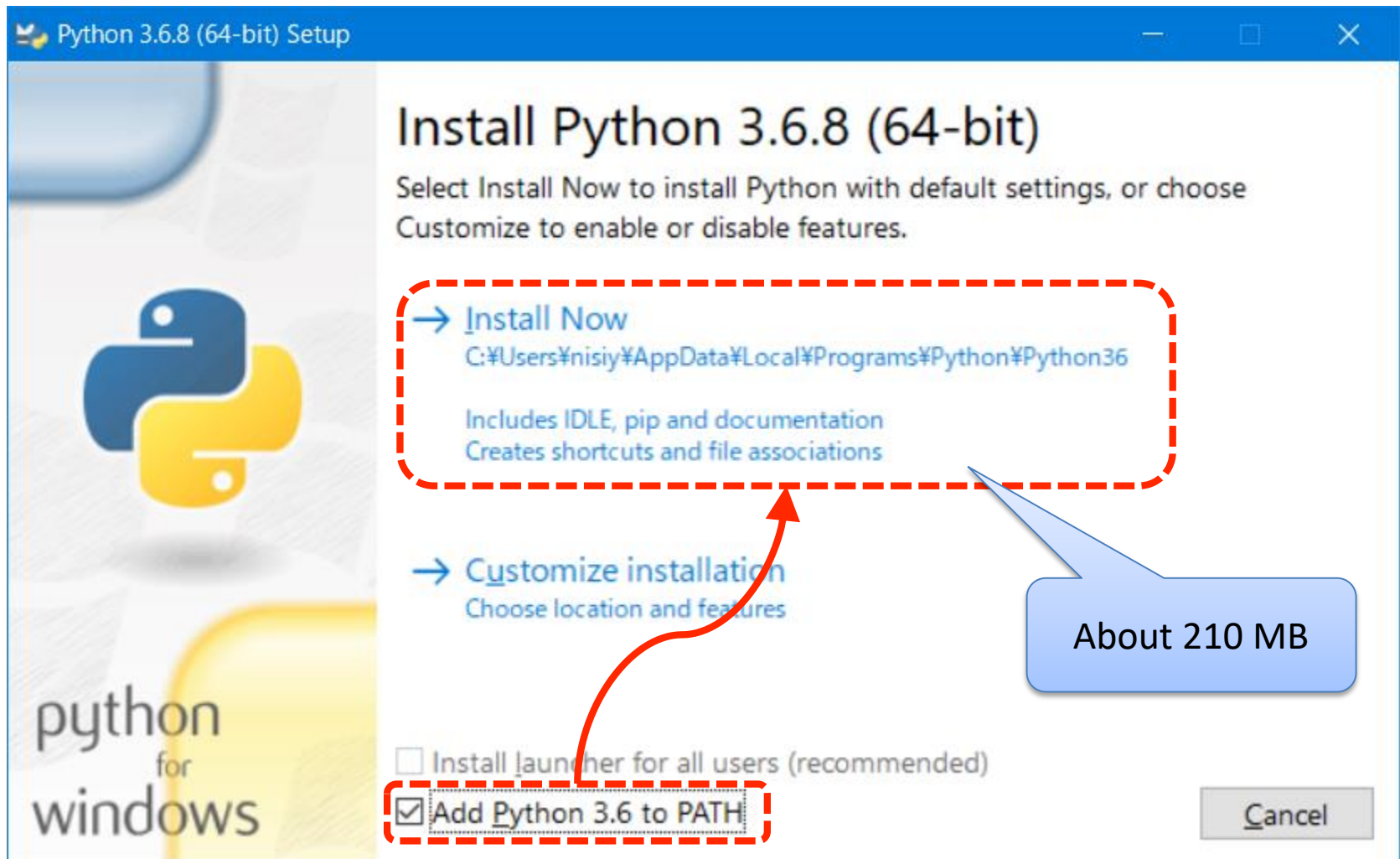
Select "Python 3.6.8"

3.7.XXX available but that can cause problems especially with GPU system.

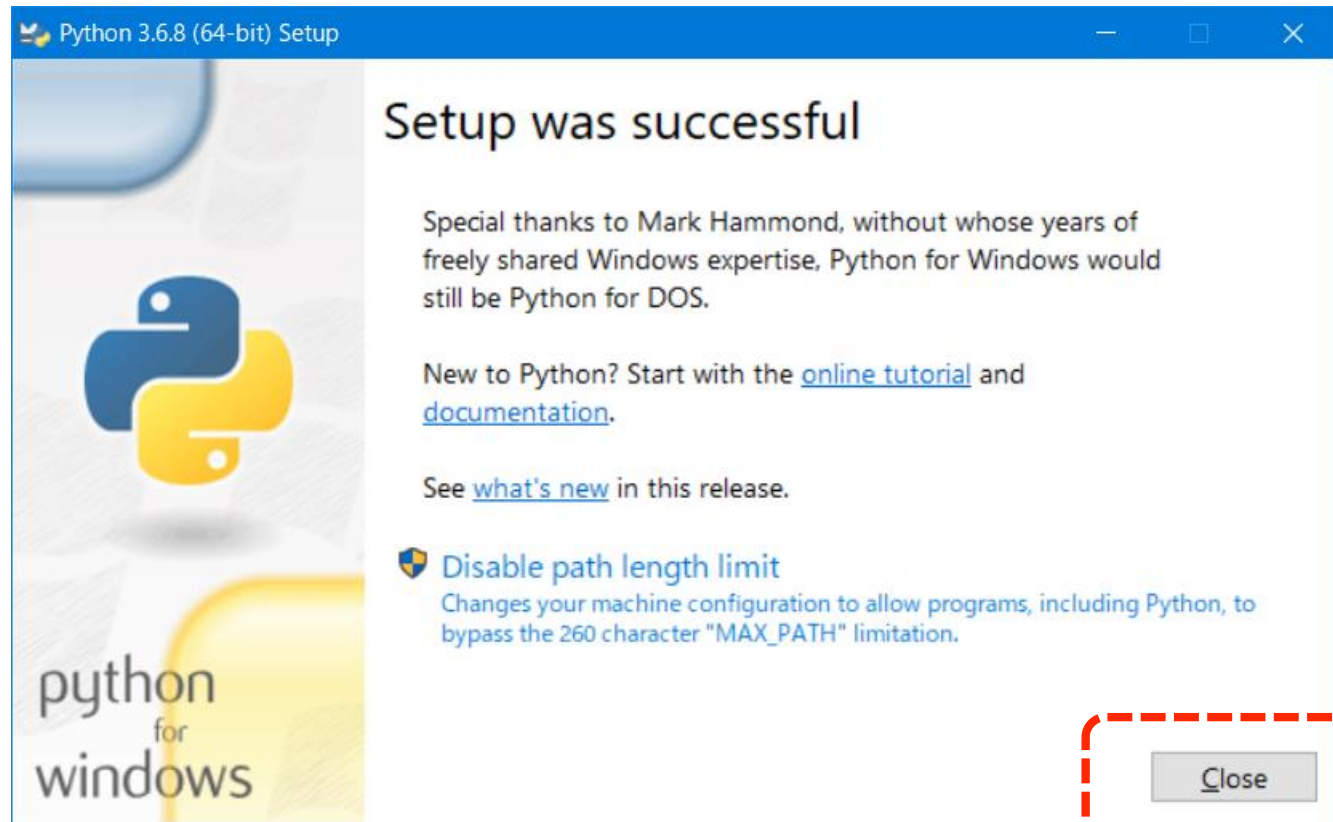
About 1.3 MB

Execute

“python-3.6.8-amd64-webinstall.exe”

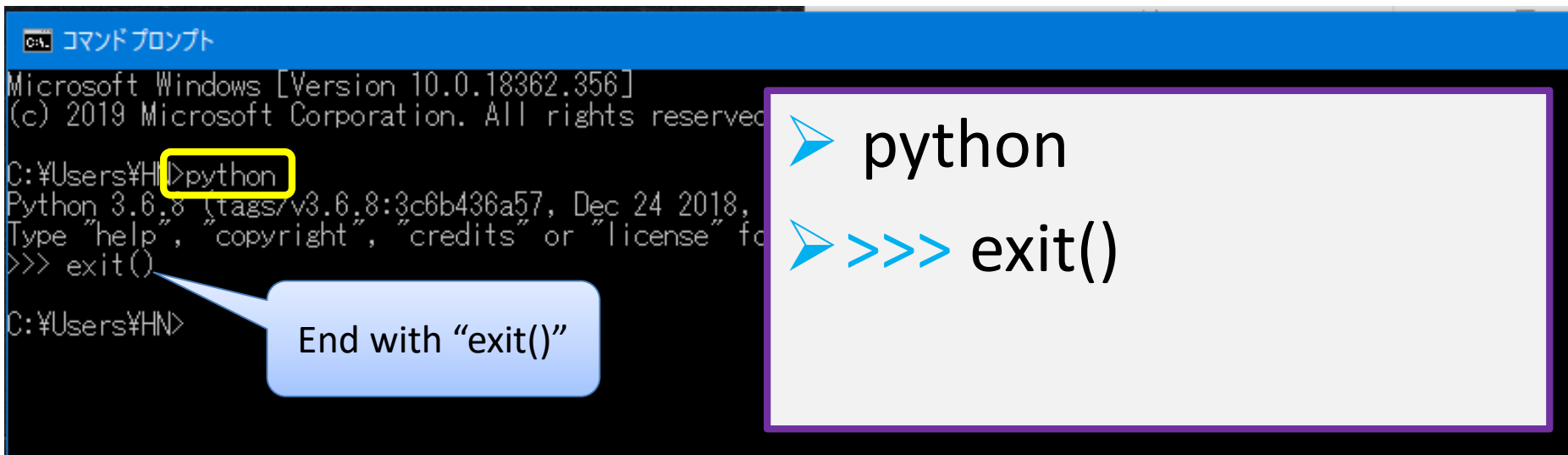


Finished installation



Check about installation

- Run the “cmd.exe” (Command prompt)
- Enter the command “python”.
- After check the short messages with version, type “exit()” command.



```

C:\> コマンド プロンプト
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\HN> python
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 2018,
Type "help", "copyright", "credits" or "license" for
>>> exit()

C:\Users\HN>

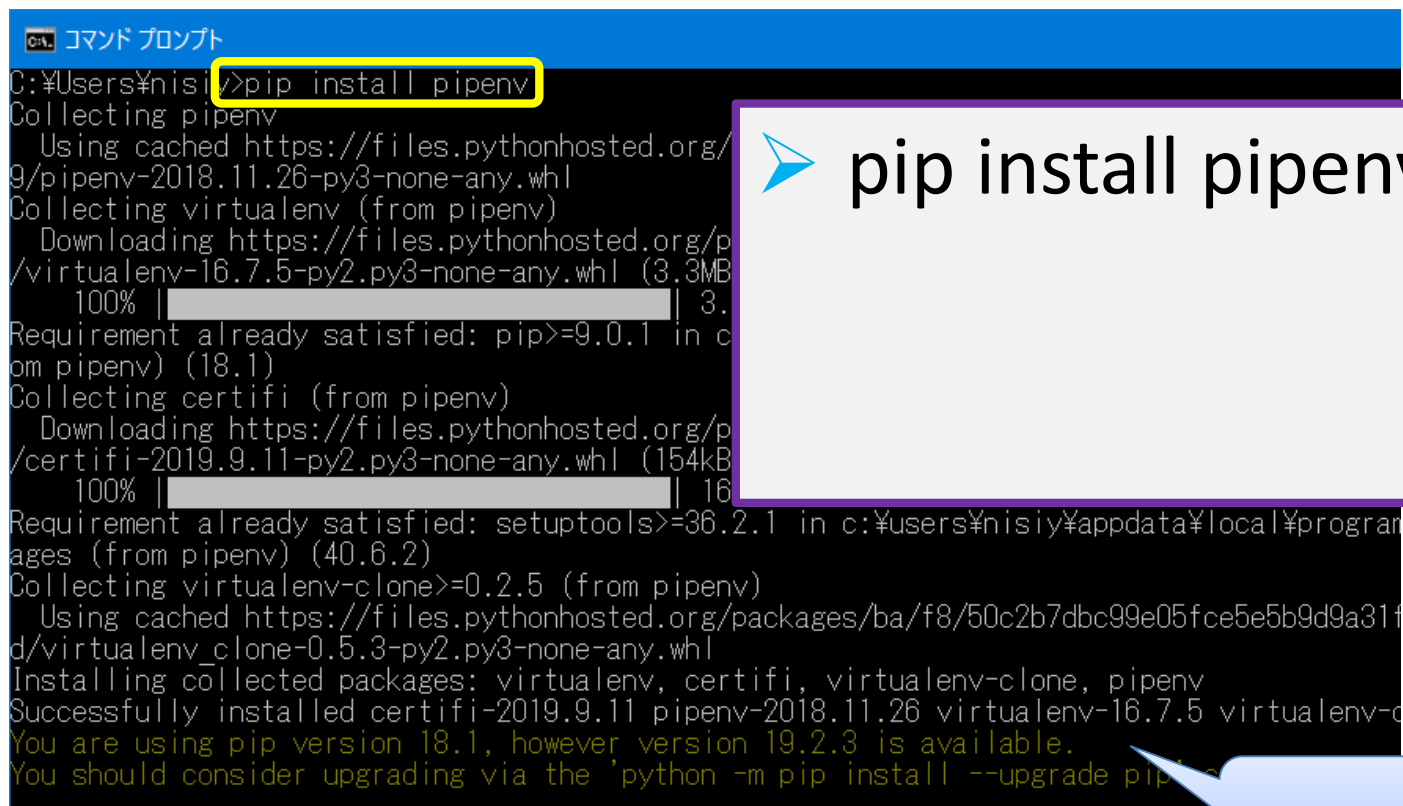
```

End with “exit()”

- python
- >>> exit()

Install “pipenv” software with “pip install” command.

- Enter the command as “pip install pipenv.”



```
コマンド プロンプト
C:\Users\nisiy>pip install pipenv
Collecting pipenv
  Using cached https://files.pythonhosted.org/9/pipenv-2018.11.26-py3-none-any.whl
Collecting virtualenv (from pipenv)
  Downloading https://files.pythonhosted.org/p/virtualenv-16.7.5-py2.py3-none-any.whl (3.3MB)
  100% |#####| 3.
Requirement already satisfied: pip>=9.0.1 in c:\users\nisiy\appdata\local\program
om pipenv) (18.1)
Collecting certifi (from pipenv)
  Downloading https://files.pythonhosted.org/p/certifi-2019.9.11-py2.py3-none-any.whl (154kB)
  100% |#####| 16
Requirement already satisfied: setuptools>=36.2.1 in c:\users\nisiy\appdata\local\program
ages (from pipenv) (40.6.2)
Collecting virtualenv-clone>=0.2.5 (from pipenv)
  Using cached https://files.pythonhosted.org/packages/ba/f8/50c2b7dbc99e05fce5e5b9d9a31f
d/virtualenv_clone-0.5.3-py2.py3-none-any.whl
Installing collected packages: virtualenv, certifi, virtualenv-clone, pipenv
Successfully installed certifi-2019.9.11 pipenv-2018.11.26 virtualenv-16.7.5 virtualenv-c
You are using pip version 18.1, however version 19.2.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

➤ pip install pipenv

About 400 MB

Make subdir and “pipenv install”

コマンド プロンプト

```
C:\Users\nisiy>mkdir db136
```

```
C:\Users\nisiy>cd db136
```

*To install python 3.6.8 correctly, run “pipenv --python 3.6.8”.
With "pipenv install" that is usually performed, there is a risk of installing "python 3.7.x" depending on the environment.

コマンド プロンプト

```
C:\Users\HN\dp136>pipenv --python 3.6.8
```

```
Creating a virtualenv for this project
```

```
Pipfile: C:\Users\nisiy\dp136\Pipfile
```

```
Using c:\Users\nisiy\AppData\Local\programs\python\python36\python.exe (3.6.8) to create the virtual environment...
```

```
[ ] Creating virtual environment...Already using interpreter c:\Users\nisiy\AppData\Local\programs\python\python36\python.exe
```

```
Using base prefix 'c:\Users\nisiy\AppData\Local\programs\python\python36\python.exe'
```

```
New python executable in C:\Users\nisiy\dp136\virtualenv\bin\python.exe
```

```
Installing setuptools, pip, wheel...
```

```
done.
```

```
Successfully created virtual environment!
```

```
Virtualenv location: C:\Users\nisiy\dp136\virtualenv
```

```
Creating a Pipfile for this project...
```

```
Pipfile.lock not found, creating...
```

```
Locking [dev-packages] dependencies...
```

```
Locking [packages] dependencies...
```

```
Updated Pipfile.lock (ca72e7)!
```

```
Installing dependencies from Pipfile.lock (ca72e7)...
```

```
===== 0/0 - 00:00:00
```

```
To activate this project's virtualenv, run pipenv shell.
```

```
Alternatively, run a command inside the virtualenv with pipenv run.
```

```
C:\Users\nisiy\dp136>
```

- mkdir [sub]
- cd [sub]
- pipenv -- python 3.6.8*

Required libraries without GPU system.

- tensorflow [1.14.0]*
- numpy [1.16.4]*
- keras
- jupyter
- scipy
- matplotlib
- pandas
- scikit-learn
- cython
- opencv-python
- h5py
- pillow
- pydicom

Total about 10 GB

*Specifying the “tensorflow” and “numpy” version is very important.

Install libraries with command of “pipenv install *[library_names list]*”

- pipenv install tensorflow==1.14.0
- pipenv install numpy==1.16.4
- ...
- pipenv install jupyter
- pipenv install matplotlib pandas
- pipenv install pydicom

You can install by specifying the library version number.

By placing multiple library names on a single line, you can install them simultaneously.

If an error occurs, see the last two pages.

Required libraries with GPU system.

- tensorflow-gpu [1.12.0]*
- numpy [1.16.4]*
- keras**
- jupyter
- scipy
- matplotlib
- pandas
- scikit-learn
- cython
- opencv-python
- h5py
- pillow
- pydicom

*Specifying the “tensorflow” and “numpy” version is very important.

**If you cannot install “keras” of version “2.2.4” and exceed “2.2.4”, you need to use “keras in tensorflow” (the version of “2.1.6-tf”). In this case, “import keras” in the Python source code must be changed to “from tensorflow import keras”, and “from keras.XXX import XXX” must be changed to “from tensorflow.keras.XXX import XXX.”

Install libraries with command of “pipenv install *[library_names list]*”

- pipenv install tensorflow-gpu==1.12.0
- pipenv install numpy==1.16.4
- ...
- pipenv install jupyter
- pipenv install matplotlib pandas
- pipenv install pydicom

You can install by specifying the library version number.

By placing multiple library names on a single line, you can install them simultaneously.

If an error occurs, see the last two pages.

For example, installation of “jupyter” library in “pipenv” environment.

コマンド プロンプト

```
C:\Users\%nisiy\%dp136> pipenv install jupyter
```

```
Installing jupyter...
```

```
Adding jupyter to Pipfile's [packages]...
```

```
Installation Succeeded
```

```
Pipfile.lock (4918c7) out of date, updating to (ca72e7)...
```

```
Locking [dev-packages] dependencies...
```

```
Locking [packages] dependencies...
```

```
Success!
```

```
Updated Pipfile.lock (4918c7)!
```

```
Installing dependencies from Pi
```

```
=====
```

```
To activate this project's virtualenv, run pipenv shell.
```

```
Alternatively, run a command inside the virtualenv with pipenv run.
```

```
C:\Users\%nisiy\%dp136>
```

➤ pipenv install jupyter

How to check the list of installed libraries

```

C:\Users\HNY\dp136>pipenv shell
Launching subshell in virtual environment...
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

(dp136-n6MCN6aI) C:\Users\HNY\dp136>pip list
Package            Version
-----
-eras              2.3.0
absl-py            0.8.0
astor              0.8.0
attrs             19.1.0
backcall          0.1.0
bleach            3.1.0
colorama           0.4.1
cyclor             0.10.0

```

With GPU

Without GPU

```

(dp136-n6MCN6aI) C:\Users\HNY\dp136>pip list
Package            Version
-----
-eras              2.3.0
absl-py            0.8.0
astor              0.8.0
attrs             19.1.0
backcall          0.1.0
bleach            3.1.0
colorama           0.4.1
cyclor             0.10.0

```

```

(dp136-wo-gpu-66WSIfDe) C:\Users\HNY\dp136-wo-gpu>exit
C:\Users\HNY\dp136-wo-gpu>

```

Execute the “pipenv shell” to enter the virtual environment in the directory where the corresponding virtual environment is built.

➤ pipenv shell

Then, execute “pip list” to list up installed packages

➤ pip list

Check the list of packages displayed. (Note that more than the installed list is output.)

After checked, do following command to exit from virtual environment.

➤ exit

* While the virtual environment is running, “(virtual environment name)” will be added at the beginning of the prompt line.

By “pipenv run jupyter notebook”,
you can start.

```
C:\Users\HNY\dp136>pipenv run jupyter notebook  
[I 20:55:00.423 NotebookApp] Serving notebooks from local directory: C:\Users\HNY\dp136  
[I 20:55:00.424 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
```

Home Page - Select or create a notebook

localhost:8888/tree

jupyter

Files Running Clusters

Select items to perform actions on them.

	Name	Last Modified	File size
☐	0		
☐	1_Classification	1ヶ月前	
☐	Tkinter-test.ipynb	1ヶ月前	3.74 kB
☐	Pipfile	1ヶ月前	324 B
☐	Pipfile.lock	6分前	49.7 kB
☐	pipList.txt	1ヶ月前	2.31 kB

➤ pipenv run jupyter notebook

For check the GPU status

- `cd C:\Program Files\NVIDIA Corporation\NVSMI`
- `nvidia-smi.exe`

Displayed CUDA version is not correct.

```

C:\> コマンドプロンプト
c:\Program Files\NVIDIA Corporation\NVSMI>nvidia-smi.exe
Sat Sep 28 19:38:49 2019

+-----+
| NVIDIA-SMI 425.25                Driver Version: 425.25          CUDA Version: 10.1         |
+-----+-----+-----+-----+-----+-----+
| GPU   Name                TCC/WDDM  Bus-Id          Disp.A   Volatile Uncorr. ECC  |
| Fan  Temp  Perf  Pwr:Usage/Cap      Memory-Usage  GPU-Util  Compute M.  |
+-----+-----+-----+-----+-----+-----+
|    0   GeForce GTX 106...  WDDM     00000000:01:00.0  On          N/A             |
| 27%   35C    P8      8W / 120W      683MiB /  6144MiB      2%         Default       |
+-----+-----+-----+-----+-----+-----+

+-----+
| Processes:                                     GPU Memory |

```

Continue to Learn the Deep Learning World.

(The next two slides is for installation errors)

If an installation error occurs, delete the library and retry...

```
コマンド プロンプト
C:\Users\nisiy\dp136>pipenv install opencv-python
Installing opencv-python...
Adding opencv-python to Pipfile's [packages]...
Installation Succeeded
Pipfile.lock (1cd370) out of date
Locking [dev-packages] dependencies...
Locking [packages] dependencies...
Locking Failed!
Traceback (most recent call last):
  File "c:\Users\nisiy\appdata\local\pipenv\patched\pip\task_install.py", line 336, in _error_message
    raise ReadTimeoutError(self._source_link_url, timeout)
pipenv.patched.pip._vendor.requests.exceptions.ReadTimeout: Read timed out.
```

- pipenv install [library]
... if installation error occurred
- pipenv uninstall [library]
... and retry...

```
コマンド プロンプト
C:\Users\nisiy\dp136>pipenv uninstall opencv-python
Uninstalling opencv-python...
Removing opencv-python from Pipfile...
Locking [dev-packages] dependencies...
Locking [packages] dependencies...
```

If an installation error occurs, delete the library and retry.
If that doesn't work, please refer to the next page.

Please check the last line of error,
such as “ReadTimeoutError”.

- In English
 - <https://github.com/pypa/pipenv/issues/3110>
 - <https://github.com/pypa/pipenv/issues/1274>
- In Japanese
 - http://pynote.hatenablog.com/entry/python_ubuntu_pipenv#pexpectexceptionsTIMEOUT-%E3%82%A8%E3%83%A9%E3%83%BC
 - <https://qiita.com/kazetof/items/b18b225b7a3143514485>