

Deep Learning 작업환경 조성 & 사용법

ISL
안재원

- Ubuntu 설치
- 작업환경 조성
- 접속 방법
- 사용 예시

Ubuntu 설치

- ISO file Download

※ www.ubuntu.com

Ubuntu for personal and mobile computing
I want convergence now!

Ubuntu for cloud computing
I want Ubuntu running my cloud and as a guest in my cloud of choice.

Ubuntu for things
I want a secure, upgradeable Internet of Things, powered by Ubuntu.

Community projects
I support LoCo teams, UbuCons and other events, upstream projects and all the good work the community does.

Tip to Canonical
Hats off for making Ubuntu possible. Keep it up.

 **The same price as**
King Kong versus Godzilla on DVD
\$15

Your contribution
\$ 15

[Not now, take me to the download >](#) [Pay with PayPal](#)

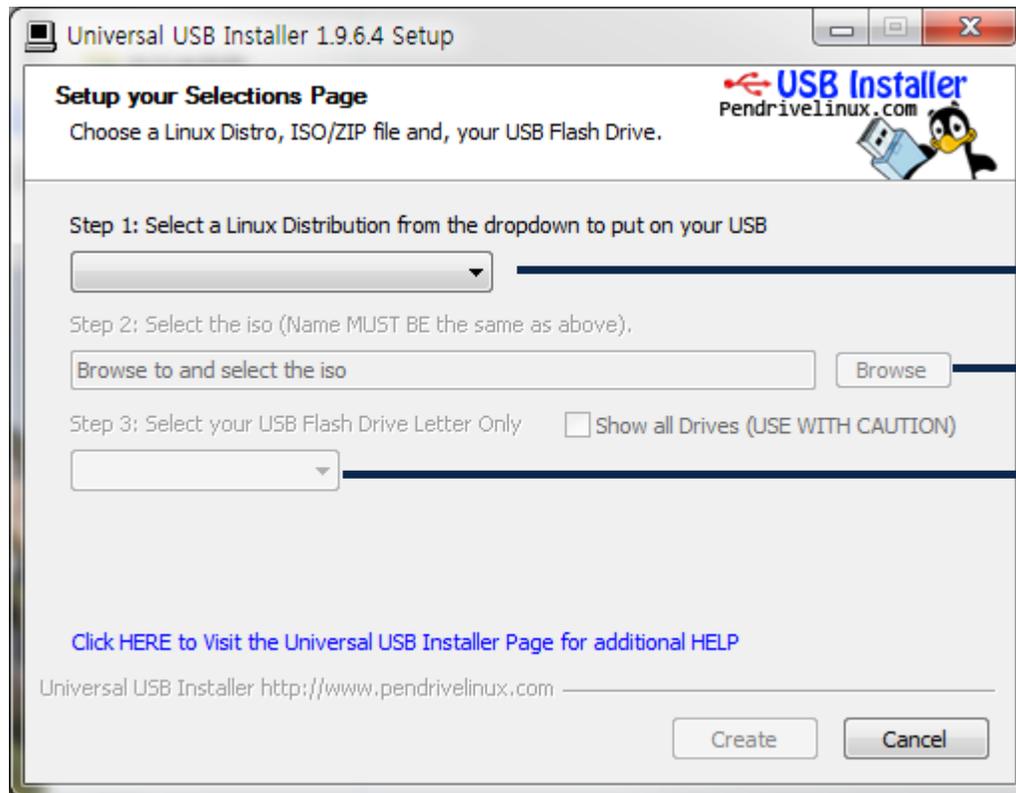
We use cookies to improve your experience. By your continued use of this site you accept such use. To change your settings please [see our policy.](#) ✕

Ubuntu 설치

01

- *Make Booting USB*

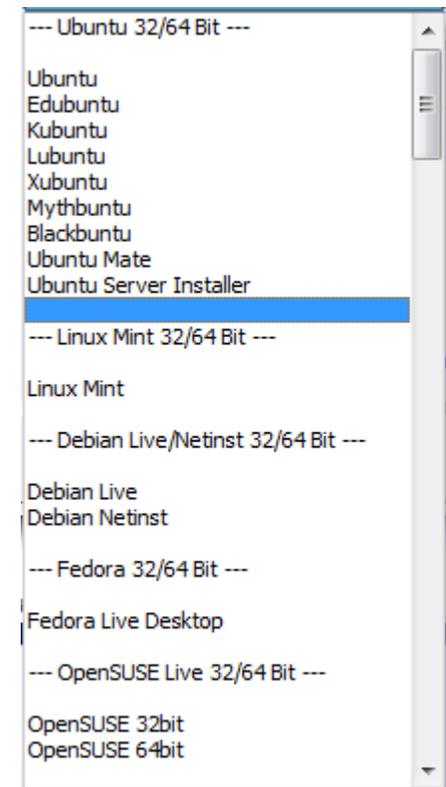
※ <http://www.pendrivelinux.com/universal-usb-installer-easy-as-1-2-3/>



Linux version

ISO file

Select Device
with Format



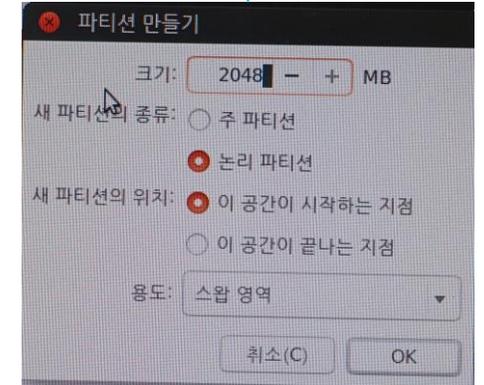
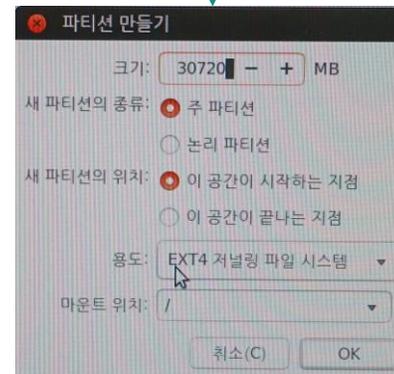
Ubuntu 설치

- Install

※ USB Booting



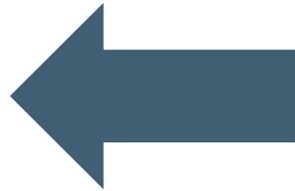
- 파티션 설정.



Ubuntu 설치

- Swap partition

※ Swap 파티션



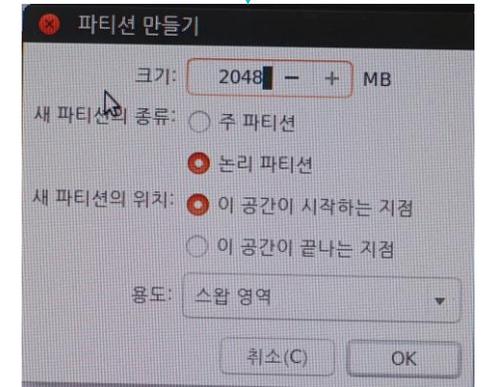
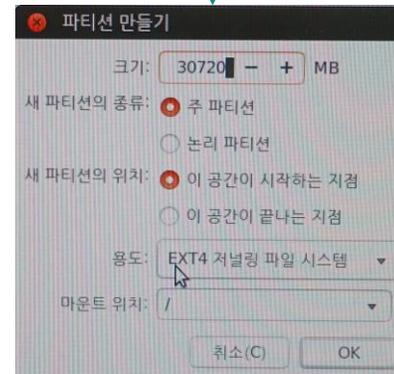
- 파티션 설정.



1. 메모리가 가득 차 프로그램을 실행할 수 없을 때.
2. 최대 절전 모드.



보조 기억장치 소모



작업환경 조성

02

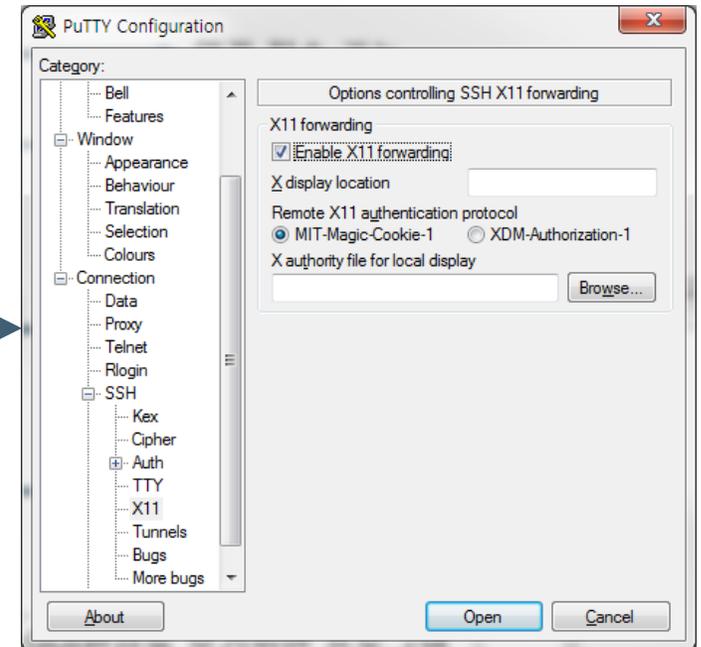
- Install list

※ Ubuntu

- 1. SSH server → 원격 접속 가능
- 2. 그래픽 카드 드라이버 → 최신버전 업데이트
- 3. CUDA(+cuDNN)
- 4. Anaconda(+ipython)
- 5. OpenCV
- 6. Caffe(+Digits) → Deep learning 프로그램
- 7. TensorFlow설치

※ Window

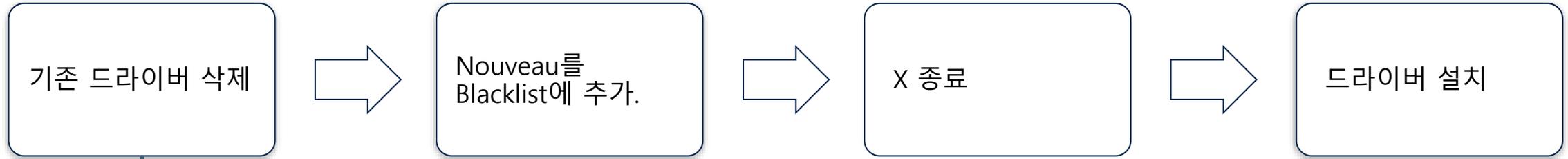
- 1. putty → 원격 접속 프로그램
- 2. Xming → GUI환경을 보기위한 프로그램



작업환경 조성

02

- 그래픽 카드 드라이버 설치

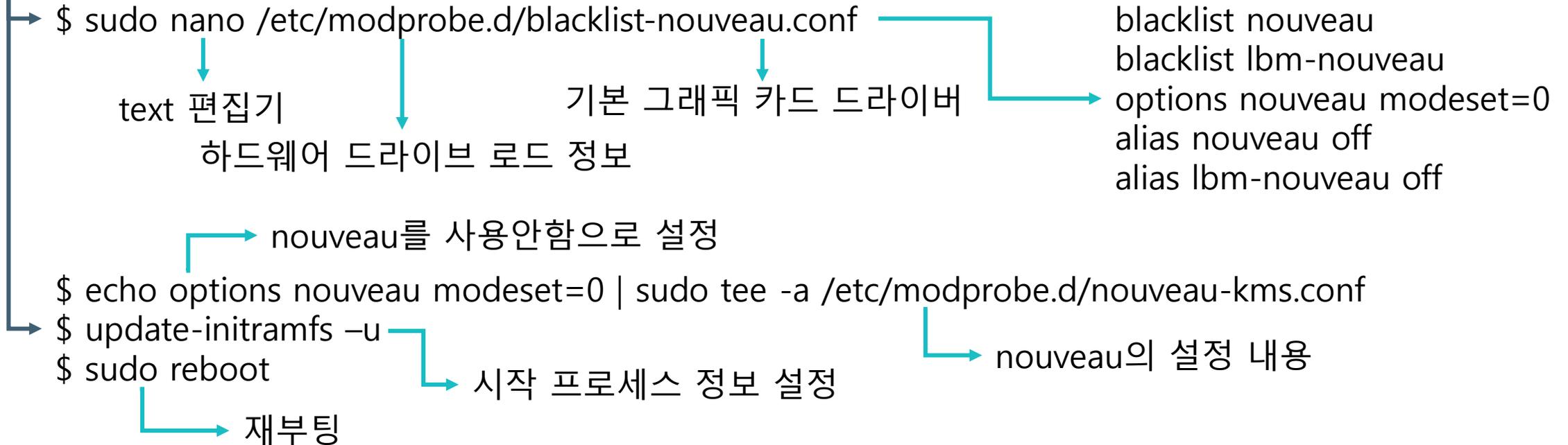
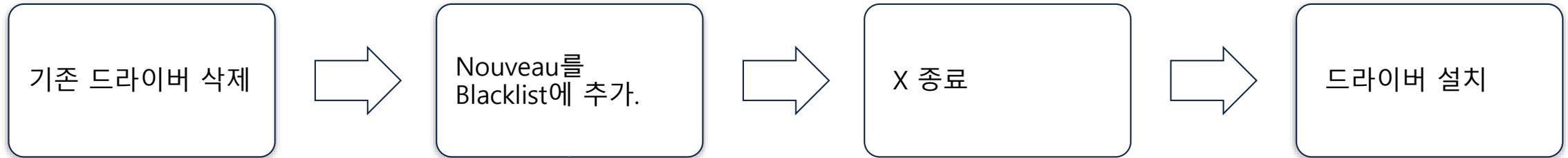


`$sudo apt-get remove nvidia* && sudo apt-get autoremove`
 ↑ 관리자 권한으로 동작
 ↑ 패키지 관리 명령
 ↓ nvidia의 모든 패키지 삭제
 ↓ 필요 없는 패키지 삭제

작업환경 조성

02

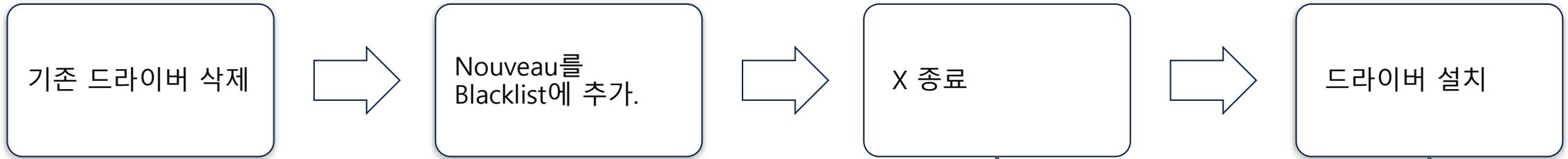
- 그래픽 카드 드라이버 설치



작업환경 조성

02

- 그래픽 카드 드라이버 설치



→ \$ service lightdm stop

└─ OS 버전마다 다름

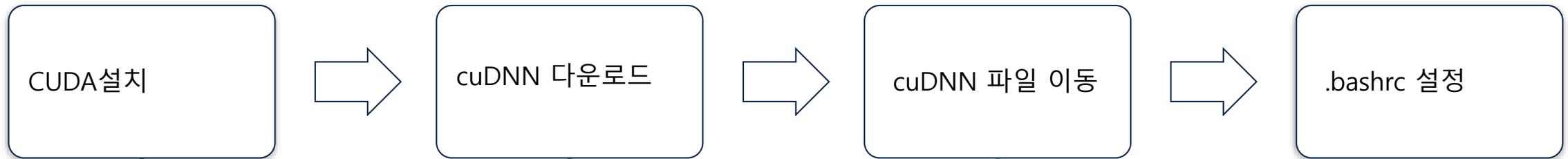
→ \$ sudo sh NVIDIA-Linux*.run

\$ sudo reboot

작업환경 조성

02

- CUDA(+cuDNN) 설치



→ Nvidia 홈페이지 에서 다운로드

※ <https://developer.nvidia.com/cuda-toolkit>

- cudnn.h → /usr/local/cuda/include
- libcudnn* → /usr/local/cuda/lib64

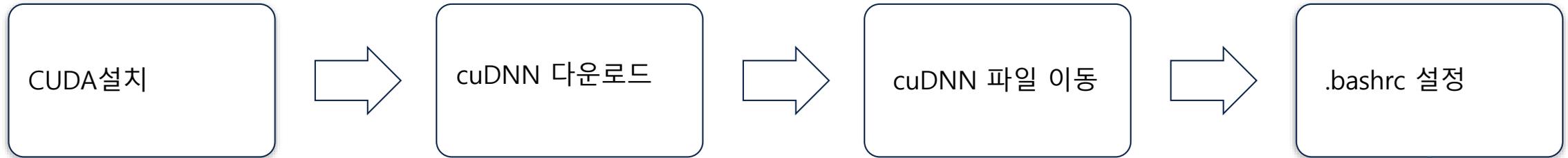
- CUDA version 7.5 → 그래픽 카드 드라이버 설치과정 스킵
- cuDNN version 4 → 최신 버전은 version 5

└─> CUDA Deep Neural Network Library

작업환경 조성

02

- CUDA(+cuDNN) 설치



- bash(Bourne-again shell)의 참고할 내용이 담긴 파일(환경변수, 사용자 지정 명령어)
- bash는 유닉스에서 사용하는 커맨드 shell
- shell은 운영체제의 기능과 서비스를 구현하는 인터페이스 제공(CLI, GUI)

```
$ cd && sudo nano .bashrc
$ source .bashrc
```

↪ 변경된 환경변수 적용

```
export PATH=/usr/local/cuda-7.5/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda-7.5/lib64:$LD_LIBRARY_PATH
```

작업환경 조성

02

- Caffe 설치

Source 다운로드

make 설정

make

```
$ git clone https://github.com/BVLC/caffe.git
```

↳ git 서버에 있는 자료를 받아 사용

↳ 파일 복사

```
$ cp Makefile.config.example Makefile.config
```

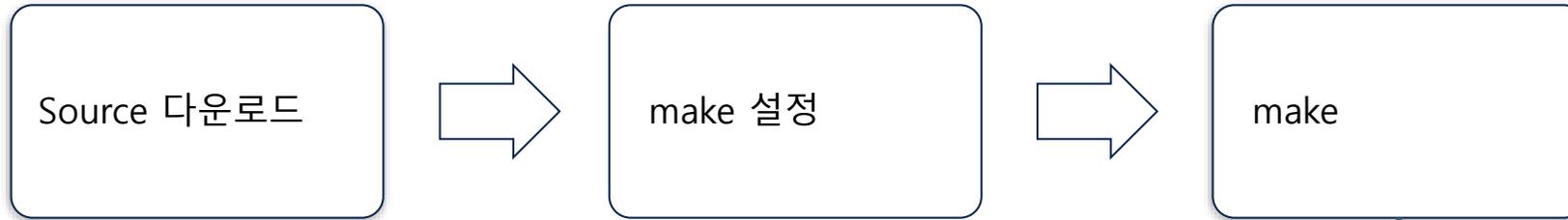
```
$ nano Makefile.config
```

↳ vi, vim, gedit 등등

```
USE_CUDNN := 1
OPENCV_VERSION := 3.1
ANACONDA_HOME := /home/isl/anaconda2
PYTHON_INCLUDE := .....
PYTHON_LIB := $(ANACONDA_HOME)/lib
WITH_PYTHON_LAYER := 1
USE_PKG_CONFIG := 1
```

작업환경 조성

- Caffe 설치



```
$ make clean  
$ sudo make all  
$ sudo make test
```

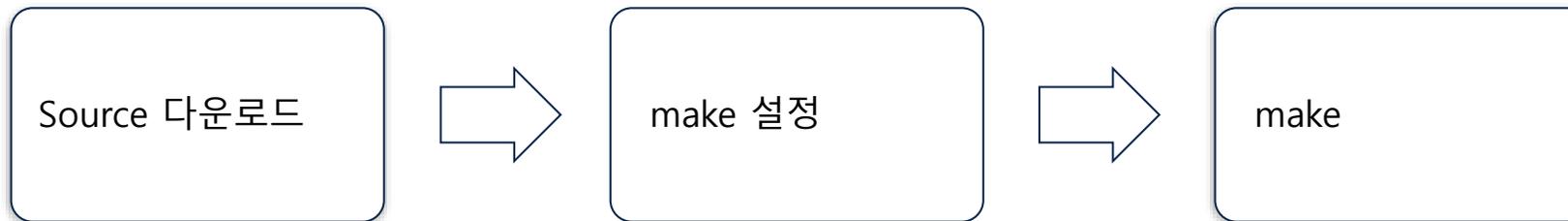
```
$ cd && sudo nano .bashrc  
$ source .bashrc
```

```
export LD_LIBRARY_PATH = $LD_LIBRARY_PATH:/home/isl/anaconda2/lib  
export PYTHONPATH=/home/isl/caffe/python:$PYTHONPATH
```

```
$ make runtest
```

작업환경 조성

- Caffe 설치



```
$ cd /caffe && make pycaffe  
$ make distribute  
$ cd /caffe/python
```

```
$ python
```

```
.....
```

```
>> import caffe
```

작업환경 조성

02

- TensorFlow 설치

※ <https://www.tensorflow.org>

TensorFlow™ GET STARTED TUTORIALS HOW TO API RESOURCES ABOUT Fork me on GitHub

Version: r0.8

Introduction

Recommended Next Steps

Download and Setup

Requirements

Overview

Pip Installation

Virtualenv installation

Anaconda environment installation

Docker installation

Test the TensorFlow installation

(Optional, Linux) Enable GPU Support

Run TensorFlow from the Command Line

Run a TensorFlow demo model

Installing from sources

Clone the TensorFlow repository

Installation for Linux

Installation for Mac OS X

Create the pip package and install

Setting up TensorFlow for Development

Train your first TensorFlow neural net model

Common Problems

Download and Setup

You can install TensorFlow either from our provided binary packages or from the github source.

Requirements

The TensorFlow Python API supports Python 2.7 and Python 3.3+.

The GPU version (Linux only) requires the Cuda Toolkit >= 7.0 and cuDNN >= v2. Please see [Cuda installation](#) for details.

Overview

We support different ways to install TensorFlow:

- **Pip install:** Install TensorFlow on your machine, possibly upgrading previously installed Python packages. May impact existing Python programs on your machine.
- **Virtualenv install:** Install TensorFlow in its own directory, not impacting any existing Python programs on your machine.
- **Anaconda install:** Install TensorFlow in its own environment for those running the Anaconda Python distribution. Does not impact existing Python programs on your machine.
- **Docker install:** Run TensorFlow in a Docker container isolated from all other programs on your machine.

If you are familiar with Pip, Virtualenv, Anaconda, or Docker, please feel free to adapt the instructions to your particular needs. The names of the pip and Docker images are listed in the corresponding installation sections.

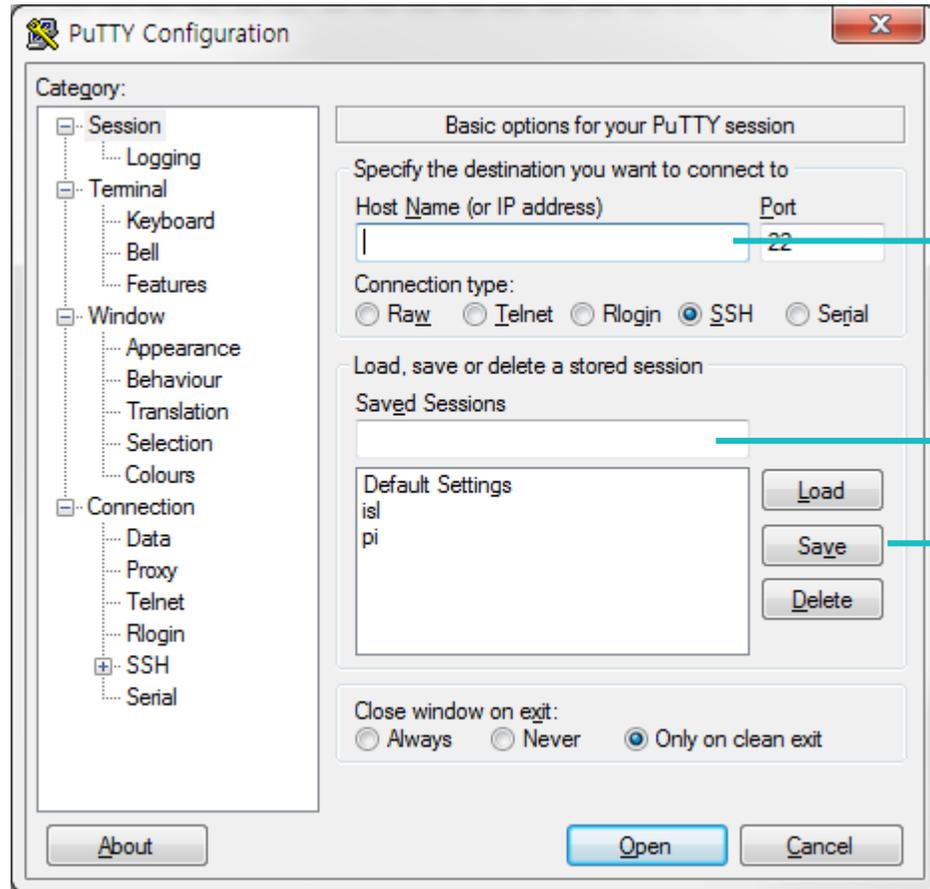
If you encounter installation errors, see [common problems](#) for some solutions.

Pip Installation

Anaconda 환경에서 사용하기 때문에

접속 방법

- putty



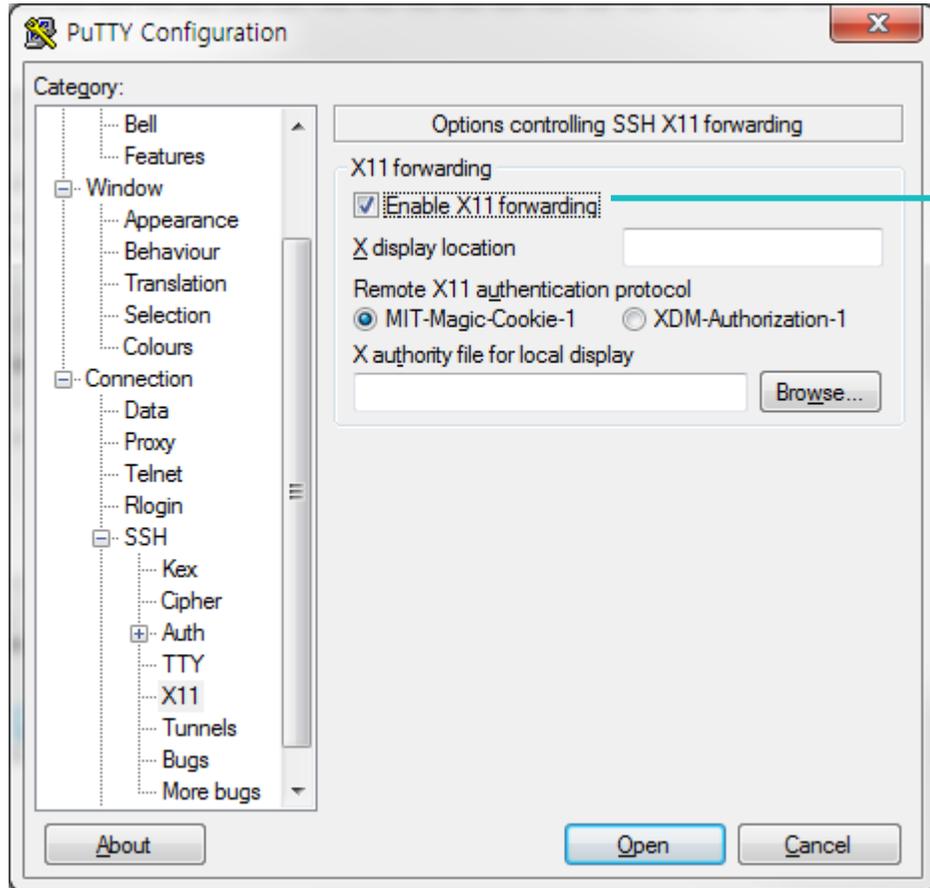
168.188.117.123

설정 저장할 이름

설정 관리

접속 방법

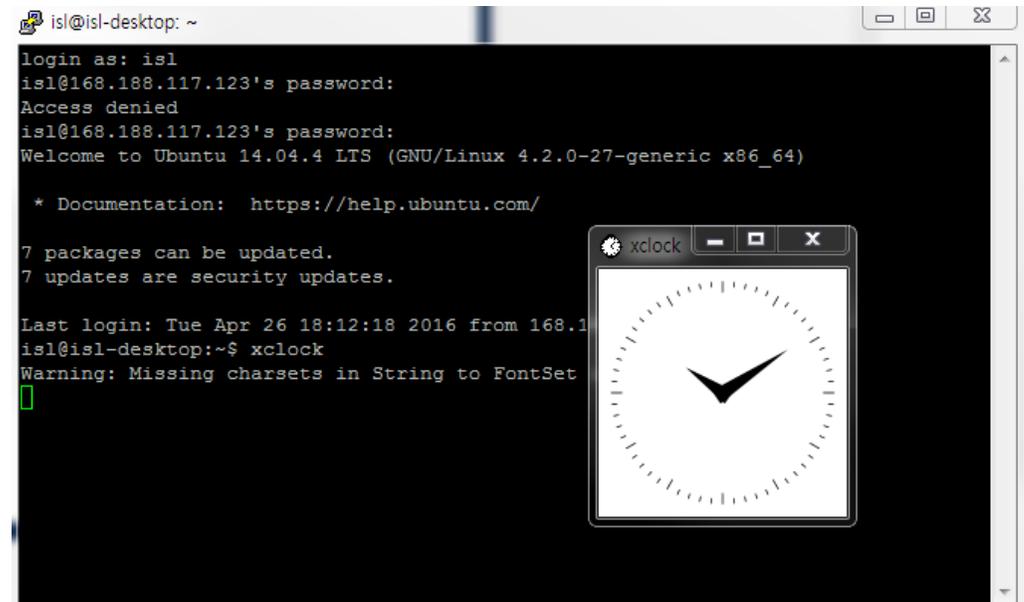
- putty



: Xming를 Window 백그라운드에 실행

활성화 설정

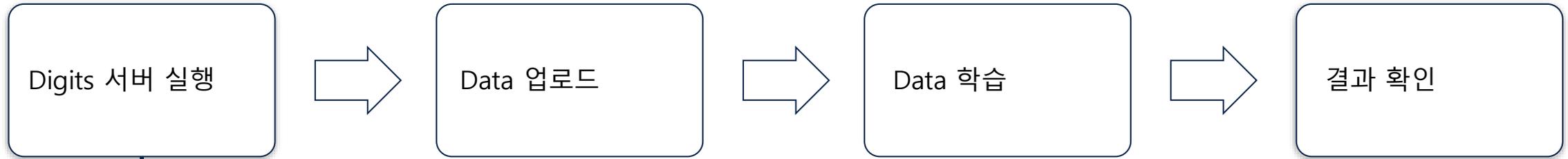
\$ xclock



사용 예시

04

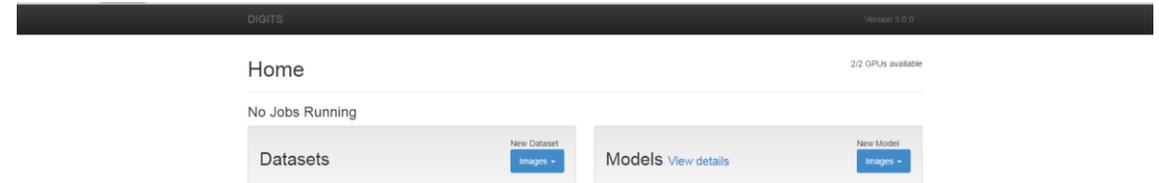
- Digits



```

$ sudo stop nvidia-digits-server
$ sudo start nvidia-digits-server
  
```

- Internet Browser를 통해 접속
(168.188.117.123)



사용 예시

04

- Digits

DIGITS Version 3.0.0

Home 2/2 GPUs available

No Jobs Running

Datasets New Dataset
Images

test	Imdb	Delete
------	------	--------

Submitted: 10:59:35 AM
Status: Done after 3 seconds

Models [View details](#) New Model
Images

- Classification
- Other

168.188.117.123/models/images/classification/new

사용 예시

- Digits

The screenshot shows the DIGITS web interface for an Image Classification Model. At the top, there is a header with 'DIGITS', 'Image Classification Model', and 'Version 3.0.0'. Below the header is a line graph showing the 'Learning Rate' over 'Epochs'. The learning rate starts at 0.01 and drops to 0.001 at epoch 10, then to 0.0001 at epoch 20. Below the graph is the 'Trained Models' section, which includes a 'Select Model' dropdown menu currently set to 'Epoch #30' and a 'Download Model' button. To the left of the 'Trained Models' section is the 'Test a single image' section, which has an 'Image URL' input field, an 'Upload image' button, and a 'Classify One' button. To the right of the 'Test a single image' section is the 'Test a list of images' section, which has an 'Upload Image List' button, a 'Classify Many' button, and a 'Number of images use from the file' input field set to 100. There are also annotations: a pink box around the 'Download Model' button with an arrow pointing to the text '학습결과 다운로드', and a blue box around the 'Image URL' field with an arrow pointing to the text '이미지 입력'.

학습결과 다운로드

이미지 입력

Q & A
