

우분투 환경에서 파이토치 GPU 연산을 위한 소프트웨어 설치 과정

지능시스템 연구실
유민형

목차

1. 우분투 설치(생략)
2. 고정 IP 설정(필요시)
3. GPU 관련 소프트웨어 설치
4. 아나콘다 설치 및 가상환경 만들기
5. 파이토치 설치
6. 점검하기

1. 우분투 설치

- 설치 과정: 생략
- 우분투 설치가 되어있는 것을 가정한다.
- 본 메뉴얼은 우분투 18.04 LTS 버전(서버 버전)에 해당하는 설명이다.

2. 고정 IP 설정

- 외부에서 접속이 가능하도록 IP 주소를 할당하기.

2.1 교내 IP 주소 신청하기

고려대 포털에서 로그인 후
 "IP 신청"에 들어가서
 IP를 요청하고
 하루 정도 기다리면
 이메일로 IP 주소를 보내준다.



KUPID Korea University Portal to Information Depository

유인형 (gogoymh) ON 로그아웃 ENGLISH 검색어를 입력하세요.

메일 | 등록/장학 | 수업 | 학적/출입 | 정보생활 | 게시판 | 커뮤니티 | 지식관리 | 연구포털

개인페이지

학사일정

- 2020학년도 제2학기 학사운영계획 수정 안내(2020.11.18.) 학사팀 2020-11-18 ~ 2020-12-31
- [재무부] 2020학년도 2학기 대학(원) 재학생 복학생 재입학생 등록 안내 재무부 2020-07-27 ~ 2020-12-31
- 2021학년도 제1학기 기개설 학생설계전공 전형 합격 및 유의사항 안내 학사팀 2020-12-29 ~ 2021-01-05
- [세종] 2021학년도 캠퍼스 간 소속변경 1단계 서류전형 합격자 안내 교무학사팀 2020-12-28 ~ 2021-01-05
- [일반대학원] 2021학년도 대학원 학사일정 대학원행정팀 2020-12-10 ~ 2022-02-28
- [일반대학원] 2020학년도 제 2학기 학위청구논문심사 주요사항 안내 대학원행정팀 2020-12-07 ~ 2021-01-06
- [일반대학원] 2021년 2월 졸업예정자 영문성명 신청 및 결정 안내 대학원행정팀 2020-12-07 ~ 2021-01-29
- [일반대학원] 2021년 2월 박사학위 취득자 설문조사 대학원행정팀 2020-12-07 ~ 2021-02-10
- 2021학년도 제1학기 이종전공 모집단위별 기준정원 안내 학사팀 2020-11-25 ~ 2020-12-30
- [일반대학원] 2020학년도 2학기 성적처리 일정 안내(학생) 대학원행정팀 2020-12-29 ~ 2021-01-06

일반 공지사항 | 규정 공지사항 | 교내의 행사 | 장학금 공지사항

- [인학개발부] 2020학년도 통계방학 중 임시휴일(12월 31일) 시행 안내 인학개발부 2020-12-28
- [11/16 업데이트] 코로나19 고려대학교 대응 지침 안내 정책기획팀 2020-11-17
- SPSS, SAS 소프트웨어 외부 사용방법 안내 정보인프라부 2020-09-31
- [KU개척마음] 메이커 스페이스 사용 가이드 KU개척마음... 2020-12-29
- 연락처 서비스기반 하나카드(개인계좌지정형 연구비 법인카드) 발급서비스 오픈 산학협력팀 2020-12-29
- [경력개발센터] 2021년 1월 모의PT/토론면접 참가자 모집 안내 경력개발센터 2020-12-29
- [도서관] 2020 CCL YouTube 경진대회 : BOOKFLEX 학술정보인프라부 2020-12-29
- [국제교류팀] 2021학년도 2학기 파견 교환학생 선발 공고 (3차 파... 국제교류팀 2020-12-29
- [인권·성평등센터] 2020 학년도 인권과 성평등 법정·의무교육 수강 안내 인권·성평등센터 2020-12-29
- [인권·성평등센터] *연장* 학부생 대상 대체강의 ~ 2020학년도... 인권·성평등센터 2020-12-29
- [인학개발부] 단속군무 안내 인학개발부 2020-12-29
- [국제대학원] 2021학년도 전기 신입생 추가모집 안내 국제대학원행정실 2020-12-29
- [기업산학협력센터] 스마트 플 활용 아이디어 공모전 기업산학연합... 2020-12-29

교육환경 개선사업

- [공사안내] 과한도서관 리모델링 2차 사업 관리자 2020-12-02
- [공사안내] 건물별 전기설비 노후로 인한 개선 공사 건축팀 2020-12-02
- [공사안내] 건물별 기계설비 노후로 인한 개선 공사 건축팀 2020-11-26

이메일에는
 IP 주소와
 gateway 주소
 DNS server 주소
 Assist DNS server 주소 등이
 적혀 있을 것이다.

2.2 IP 입력하기

- 컴퓨터의 이더넷 이름 확인하기

```
$ ifconfig
```

-> enp, eth 등으로 시작하는 이름이 있을 것이다.

- (18.04 기준) netplan 디렉토리로 이동 후 파일 확인(*우분투 버전마다 다르니 netplan 이용 방법이 맞는지 확인해 봐야 한다.)

```
$ cd /etc/netplan
```

```
$ ls
```

-> .yaml 확장자의 파일이 있을 것이다.

2.2 IP 입력하기

- .yaml 파일을 수정하기 위해 파일을 열어준다.

```
$ sudo vi 01-netcfg.yaml
```

- i 를 눌러서 파일 내용을 수정한다. 할당 받은 IP 주소를 입력해준다.

앞에서 확인한 이더넷 이름

DHCP를 no로 설정

gateway 주소를 입력한다.

```
# This file describes the network interfaces available on
# For more information, see netplan(5)
network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      addresses: [163.152.162.220/24]
      gateway4: 163.152.162.1
      nameservers:
        addresses: [163.152.1.1,163.152.11.6]
    #eth1:
    #dhcp4: no
    #addresses: [0.0.0.0/24]
    #gateway4: 0.0.0.0
    #nameservers:
    #addresses: [8.8.8.8,0.0.0.0]
```

DNS 서버 주소와
Assist DNS 서버 주소를
차례로 입력한다.

2.2 IP 입력하기

- Esc 키를 누른 후 :wq를 입력하여 저장해준다.
(sudo 권한이 없으면 저장이 안되니 :q!로 나간 후 다시 한다.)

- 고정 IP를 적용한다.

```
$ sudo netplan apply
```

- IP가 잘 적용되었는지 확인하기

```
$ hostname -I
```

-> 내 아이피 주소가 출력된다.

2.2 고정 IP 설정 시 유의 사항

- IP 입력을 마친 후에는 재부팅을 한번 해준다.
- 고정 IP 설정을 했는데 원격 접속이 안되는 경우(경험한 원인들)
 - 1) 랜선 포트가 물리적으로 연결이 제대로 안된 경우
 - > 의외로 가장 흔한 문제
 - > 컴퓨터의 다른 랜포트로 연결해서 원격 접속해보기
 - > 스위치 허브나 벽의 다른 랜포트로 연결해서
원격 접속해보기, 공유기는 스위치 허브 모드로 설정
 - 2) 보통 SSH를 통해 원격 접속을 이용하는데,
SSH 패키지 자체가 설치가 안된 경우 -> SSH 설치하기

2.2 고정 IP 설정 시 유의 사항

- 3) SSH가 부팅 후 자동 실행이 안되는 경우
-> 부팅 후 자동으로 실행되도록 설정하기
- 4) 방화벽이 설정된 경우
-> SSH의 방화벽을 해제
- 5) SSH는 22번 포트를 사용하는데 22번 포트가 닫힌 경우
-> 22번 포트를 열어준다.
- 6) 부팅 후 일정 시간이 지나지 않은 경우
-> 부팅 후 일정 시간(ex 120초)이 지나야
원격 접속이 가능하게 설정된 경우
설정 변경 혹은 기다리기

3. GPU 관련 소프트웨어 설치하기

- GPU를 딥러닝에 이용하기 위해서
NVIDIA DRIVER와 CUDA toolkit를 설치해야 한다.
- 설치 전 두 소프트웨어간 compatibility를 먼저 확인해야 한다.

3.1 Compatibility 확인하기

- <https://www.nvidia.com/Download/index.aspx>
- 위 주소에 접속하여 GPU에 맞는 driver 버전을 먼저 찾아 설치 후 그에 맞는 cuda toolkit 버전을 설치하는 것이 순서이지만 파이토치에 맞는 버전부터 반대로 찾아야한다.
- <https://pytorch.org/>
- 위 주소에 접속하여 설치 가능한 파이토치 버전을 찾는다.

3.1 Compatibility 확인하기

www.pytorch.org에서 CUDA 버전을 확인해준다.

프로젝트간 라이브러리 충돌 방지를 위해 아나콘다 파이썬 사용을 권장한다.

PyTorch Build	Stable (1.7.1)			Preview (Nightly)	
Your OS	Linux		Mac	Windows	
Package	Conda		Pip	LibTorch	Source
Language	Python			C++ / Java	
CUDA	9.2	10.1	10.2	11.0	None

Run this Command:

```
NOTE: Python 3.9 users will need to add '-c=conda-forge' for installation  
conda install pytorch torchvision torchaudio cudatoolkit=11.0 -c pytorch
```

리눅스/아나콘다/파이썬을 사용한다면 cuda toolkit은 11.0 버전까지만 가능하다.

3.1 Compatibility 확인하기

- <https://docs.nvidia.com/deploy/cuda-compatibility/index.html>
- 위 주소에 들어가면 cuda toolkit과 nvidia driver의 compatibility를 알 수 있다.

Table 1. CUDA Toolkit and Compatible Driver Versions

CUDA Toolkit	Linux x86_64 Driver Version
CUDA 11.1 (11.1.0)	>= 450.80.02
CUDA 11.0 (11.0.3)	>= 450.36.06
CUDA 10.2 (10.2.89)	>= 440.33
CUDA 10.1 (10.1.105)	>= 418.39
CUDA 10.0 (10.0.130)	>= 410.48
CUDA 9.2 (9.2.88)	>= 396.26
CUDA 9.1 (9.1.85)	>= 390.46
CUDA 9.0 (9.0.76)	>= 384.81
CUDA 8.0 (8.0.61 GA2)	>= 375.26
CUDA 8.0 (8.0.44)	>= 367.48
CUDA 7.5 (7.5.16)	>= 352.31
CUDA 7.0 (7.0.28)	>= 346.46

앞서 파이토치가 cuda toolkit 11.0 버전까지 출시 된 것을 확인했다.
11.1 버전을 이용할 수 없다는 점을 유의한다.

파이토치를 사용하기 위해선 CUDA 11.0을 사용해야하고 그에 맞는 드라이버 버전은 450.36.06 이상을 사용해야 한다.

3.2 NVIDIA Driver 설치하기

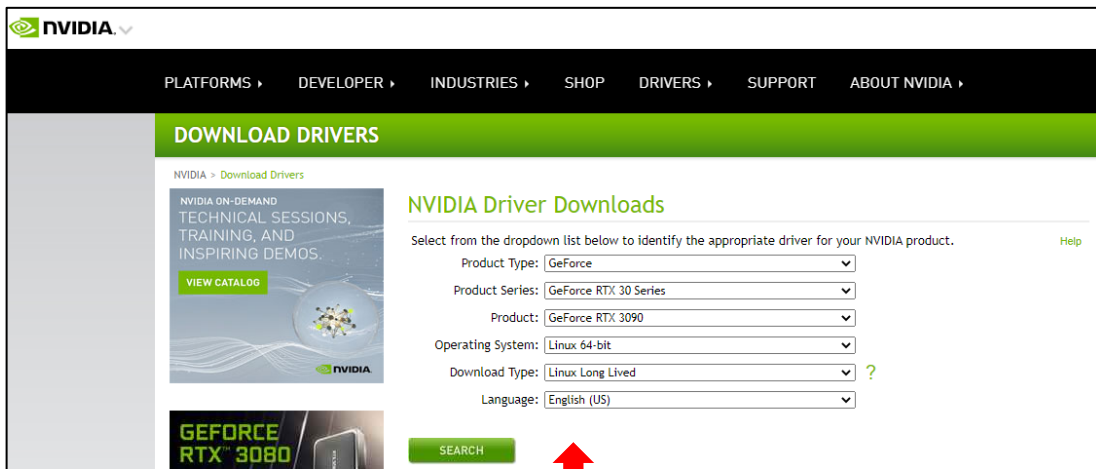
- NVIDIA DRIVER 설치를 위한 공식 가이드 주소는 다음과 같다.

<https://docs.nvidia.com/datacenter/tesla/tesla-installation-notes/index.html>

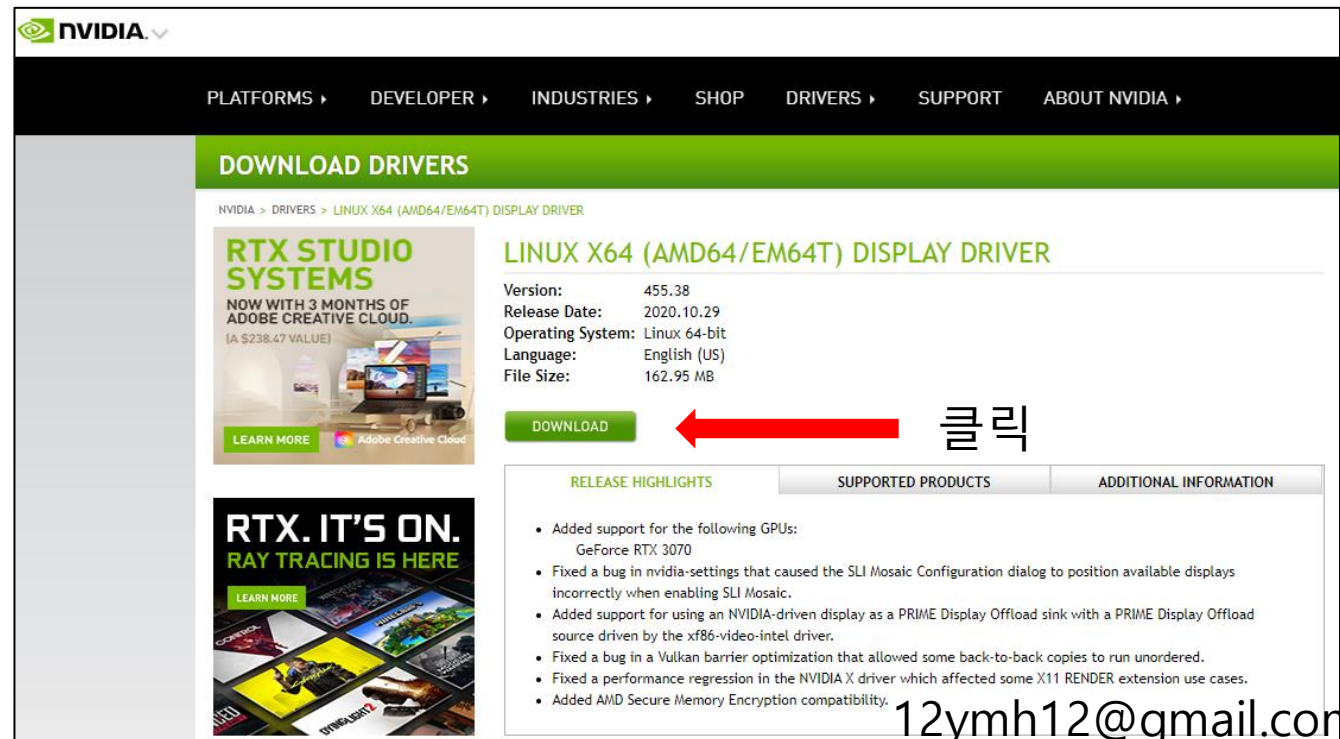
- 여러 설치 방식이 있는데 Runfile을 다운받아 실행하는 방법이 가장 간단하다.

3.2 NVIDIA Driver 설치하기

- <https://www.nvidia.com/Download/index.aspx>
- 앞서 찾은 드라이버 버전을 기억한 후 위 주소에 접속하여 GPU 모델과 OS에 맞는 버전을 검색한다.



GPU 모델과 OS 입력



3.2 NVIDIA Driver 설치하기

The screenshot shows the NVIDIA website's 'Download Drivers' page. The page has a navigation bar with links for PLATFORMS, DEVELOPERS, INDUSTRIES, SHOP, DRIVERS, SUPPORT, and ABOUT NVIDIA. Below the navigation bar is a green header for 'DOWNLOAD DRIVERS'. The main content area includes a sidebar with promotional banners for 'RTX STUDIO SYSTEMS' and 'RTX. IT'S ON. RAY TRACING IS HERE'. The main content area features a 'Download' section with a 'DOWNLOAD' button. A context menu is open over the button, and a red arrow points to the '링크 주소 복사(E)' option.

다운로드 링크 확인

3.2 NVIDIA Driver 설치하기

- 다운로드 링크를 확인해 wget을 이용해 runfile을 다운받는다.

```
$ sudo wget https://us.download.nvidia.com/XFree86/Linux-x86_64/455.38/NVIDIA-Linux-x86_64-455.38.run
```

- runfile을 실행해 설치해준다.

```
$ sudo sh NVIDIA-Linux-x86_64-455.38.run
```

3.2 NVIDIA Driver 설치하기

- 설치가 끝나면 재부팅해준다.
- 이제 nvidia-smi 명령어를 통해 GPU 상태를 확인할 수 있다.

\$ nvidia-smi

Driver 버전이 여기 보인다.

```
Tue Dec 29 16:25:15 2020
+-----+
| NVIDIA-SMI 455.38      | Driver Version: 455.38      | CUDA Version: 11.1      |
+-----+-----+-----+
| 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   GeForce RTX 3090     Off          | 00000000:19:00.0 Off  |          N/A         |
| 34%  61C   P0      111W / 350W | 0MiB / 24268MiB |    0%      Default  |
|                               |                      |              N/A         |
+-----+-----+-----+
| 1   GeForce RTX 3090     Off          | 00000000:1B:00.0 Off  |          N/A         |
| 32%  65C   P0      79W / 350W | 0MiB / 24268MiB |    3%      Default  |
|                               |                      |              N/A         |
+-----+-----+-----+
| Processes:                 |
| GPU  GI    CI             PID  Type  Process name          | GPU Memory |
|   ID  ID  ID                 |   ID           |           | Usage        |
+-----+-----+-----+
| No running processes found |
+-----+-----+-----+
```

장착한 GPU 개수와
맞는지 확인해준다.
개수가 부족하면
접촉 불량이다.
경우에 따라
GPU 수리가
필요할 수도 있다.

여기 보이는
CUDA Version은
CUDA toolkit 버전이
아니다.

CUDA toolkit이 설치
된 것으로 착각할 수
있으니 유의할 것.
드라이버 455.38 버전
은 CUDA toolkit 11.1
버전까지 설치할 수
있다는 뜻이다.

(Compuworks 설치
기사님 설명)

12ymh12@gmail.com

3.3 CUDA toolkit 설치하기

- 앞서 파이토치 사용을 위해 CUDA toolkit 11.0 버전을 설치해야 하는 것을 확인했다.
- <https://developer.nvidia.com/cuda-11.0-download-archive>
- 위 주소에 들어가 컴퓨터의 조건을 지정해준다.

3.3 CUDA toolkit 설치하기

The screenshot shows the NVIDIA Developer website's 'Select Target Platform' page. The page is titled 'Select Target Platform' and includes a green header with the NVIDIA logo and navigation links: HOME, BLOG, NEWS, FORUMS, DOCS, DOWNLOADS, TRAINING, and ACCOUNT. Below the header, there is a section with the text: 'Click on the green buttons that describe your target platform. Only supported platforms will be shown. By downloading and using the software, you agree to fully comply with the terms and conditions of the CUDA EULA.'

The selection process is organized into several categories:

- Operating System:** Windows, Linux (selected)
- Architecture:** x86_64 (selected), ppc64le, sbsa
- Distribution:** OpenSUSE, RHEL, CentOS, SLES, Ubuntu (selected)
- Version:** 20.04, 18.04 (selected), 16.04
- Installer Type:** runfile (local), deb (local) (selected), deb (network)

Below the selection area, there is a section titled 'Download Installer for Linux Ubuntu 18.04 x86_64'. It states: 'The base installer is available for download below.' Underneath, there is a 'Base Installer' section with 'Installation Instructions:' and a terminal window showing the following commands:

```
$ wget http://developer.download.nvidia.com/compute/cuda/11.0.2/local_installers/cuda_11.0.2_450.51.05_linux.run
$ sudo sh cuda_11.0.2_450.51.05_linux.run
```

조건을 지정해주면
명령어를 알려준다.



3.3 CUDA toolkit 설치하기

- 명령어를 이용해 runfile을 다운받고 실행해준다.

```
$ wget
```

```
http://developer.download.nvidia.com/compute/cuda/11.0.2/local_installers/cuda_11.0.2_450.51.05_linux.run
```

```
$ sudo sh cuda_11.0.2_450.51.05_linux.run
```

- 설치가 완료되면 재부팅해준다.

3.3 CUDA toolkit 설치하기

- 주의할 점

CUDA toolkit을 설치할 때, 최적화된 driver를 함께 설치할지 물어본다. 이미 driver를 설치한 경우 해당 옵션을 체크 해제해야 한다. 이 단계에서 다시 설치하여 충돌이 나는 경우가 많다.

- CUDA toolkit으로 driver를 한번에 설치해도 되지 않는가?

-> 된다.

그런데 가끔 알 수 없는 부분에서 충돌이 되어 원할한 이용(파이토치 GPU 연산)이 안 되는 경우가 있다.

3.4 cudnn 설치하기(옵션)

- cudnn은 딥러닝 연산 속도를 올려준다. 없어도 돌아는 간다.
- <https://developer.nvidia.com/cudnn>
- 위 주소에 들어가 CUDA toolkit 버전에 맞는 버전을 다운로드 한다. 회원가입이 필요하다.
- <https://docs.nvidia.com/deeplearning/cudnn/install-guide/index.html>
- 위 주소는 cudnn 공식 설치 가이드 문서이다.
- 여러 방식이 있는데 tar 파일을 받아 압축을 풀어서 파일을 복사해주는 방식이 가장 쉽다.

3.4 cudnn 설치하기(옵션)

- 다운로드한 파일을 압축을 풀어준다.

```
$ tar -xzvf cudnn-11.0-linux-x64-v8.0.4.30.tgz
```

- 파일을 지정된 경로로 복사해준다. 끝.

```
$ sudo cp cuda/include/cudnn*.h /usr/local/cuda/include
```

```
$ sudo cp cuda/lib64/libcudnn* /usr/local/cuda/lib64
```

```
$ sudo chmod a+r /usr/local/cuda/include/cudnn*.h  
/usr/local/cuda/lib64/libcudnn*
```

3.4 cudnn 설치하기(옵션)

- 설치 확인

```
$ cat /usr/local/cuda-11.0/include/cudnn_version.h | grep  
CUDNN_MAJOR -A 2
```

- 아래와 같이 출력되면 설치된 버전을 확인할 수 있다.

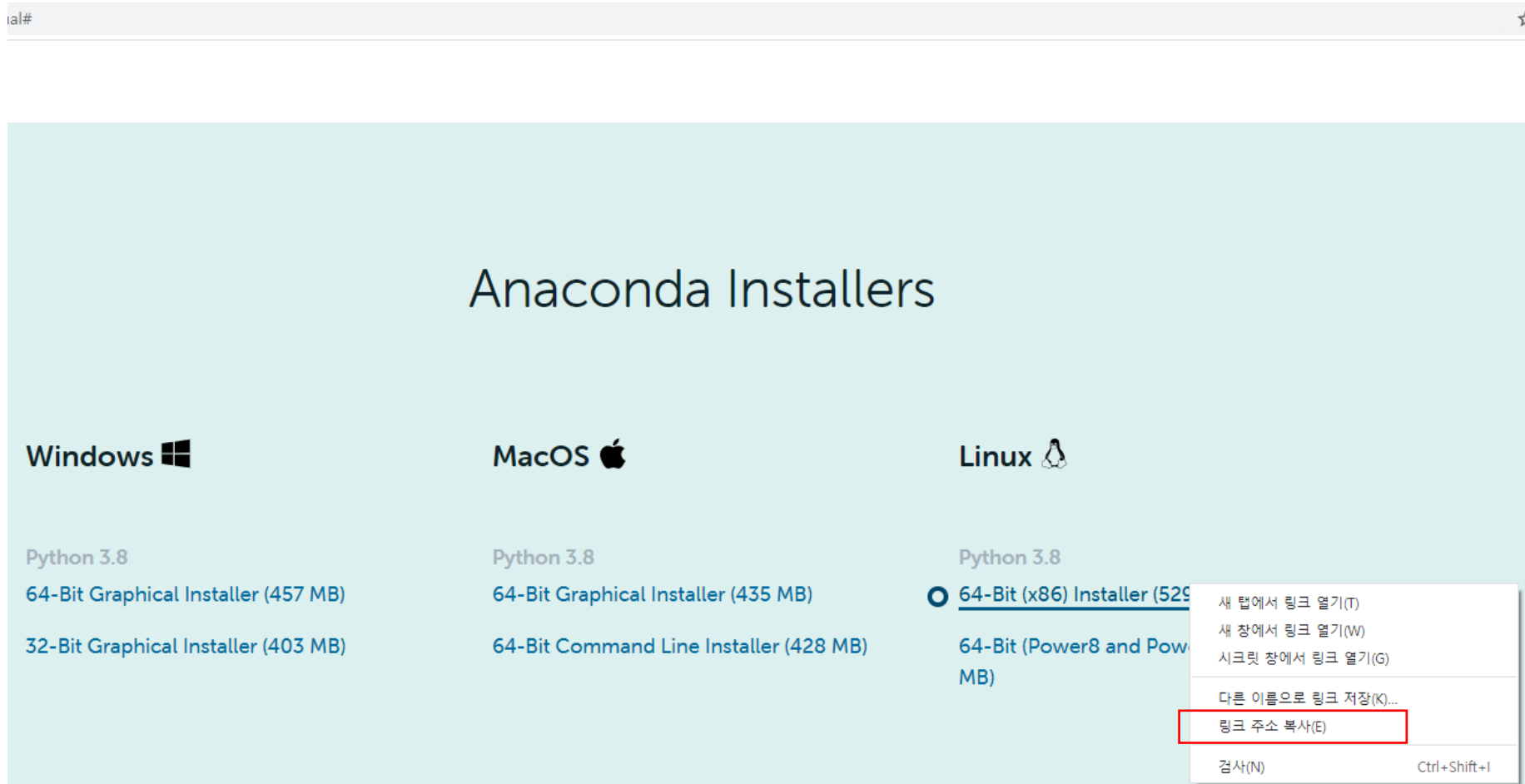
```
(base) root@ubuntu:~# cat /usr/local/cuda-11.0/include/cudnn_version.h | grep CUDNN_MAJOR -A 2  
#define CUDNN_MAJOR 8  
#define CUDNN_MINOR 0  
#define CUDNN_PATCHLEVEL 4  
--  
#define CUDNN_VERSION (CUDNN_MAJOR * 1000 + CUDNN_MINOR * 100 + CUDNN_PATCHLEVEL)  
#endif /* CUDNN_VERSION_H */
```

- 설치 후 재부팅한다.

4. 아나콘다 설치 및 가상환경 만들기

- 여러 프로젝트와 사용자 사이의 충돌을 방지하기 위해 가상환경을 만들어 사용하는 것을 권장한다.
- <https://www.anaconda.com/products/individual#>
- 위 주소에 접속해 아나콘다 오픈소스 버전을 다운로드 한다.

4.1 아나콘다 설치



주소를 복사하여
wget으로 다운로드 한다. lymh12@gmail.com

4.1 아나콘다 설치

- 주소를 복사하여 wget으로 다운로드하고 실행한다.

```
$ sudo wget https://repo.anaconda.com/archive/Anaconda3-2020.11-Linux-x86_64.sh
```

```
$ sudo sh Anaconda3-2020.11-Linux-x86_64.sh
```

- 설치 후 재부팅하면 터미널에 기본 가상환경 (base)이 뜬다.

```
(base) root@ubuntu:~# █
```

4.2 가상환경 만들기

- 프로젝트를 위해 가상환경을 만들어 준다.
- 터미널에 conda 명령어를 이용해 이름을 짓고, 파이썬 버전을 지정해 가상환경을 만들어준다.

```
$ conda create --name gpu_test python=3.8.5
```

- 기타 다른 라이브러리 설치 시에도 conda install 혹은 conda forge 등을 이용해야한다.

5. 파이토치 설치

- 가상환경에 파이토치를 설치한다.

- 가상환경 실행

```
$ conda activate gpu_test
```

- 앞서 3.1장의 <https://www.pytorch.org>에서 확인한 cuda toolkit 11.0에 맞는 명령어를 입력한다.

```
$ conda install pytorch torchvision torchaudio cudatoolkit=11.0  
-c pytorch
```

5. 파이토치 설치


- 설치가 완료되면 파이썬을 실행해서 파이토치로 GPU를 이용 가능한지 확인해준다.

\$ python

>>> import torch

>>> print(torch.cuda.is_available())  GPU를 이용할 수 있는지 체크

True

>>> print(torch.cuda.device_count())  GPU 개수가 몇 개인지 체크
여기서 개수가 다른 경우도 있다.
다르다면 접촉 불량일 가능성이 높고,
다시 장착했을 때, kernel과 라이브러리가
충돌이 발생할 가능성이 있어,
재설치를 해야 할 수도 있다.

2

```
(gpu_test) root@ubuntu:~# python
Python 3.8.5 (default, Sep  4 2020, 07:30:14)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> print(torch.cuda.is_available())
True
>>> print(torch.cuda.device_count())
2
>>> exit()
```


6. 점검하기

- 간단한 스크립트를 만들어 학습이 되는지 체크한다.

Ex) Resnet56으로 CIFAR10을 학습해보기

- 학습을 돌리고 새로운 터미널을 열어 GPU 상태를 살펴본다.

```
$ watch -n 1 nvidia-smi
```

- Single GPU와 Multiple GPU가 모두 되는지 체크해본다.