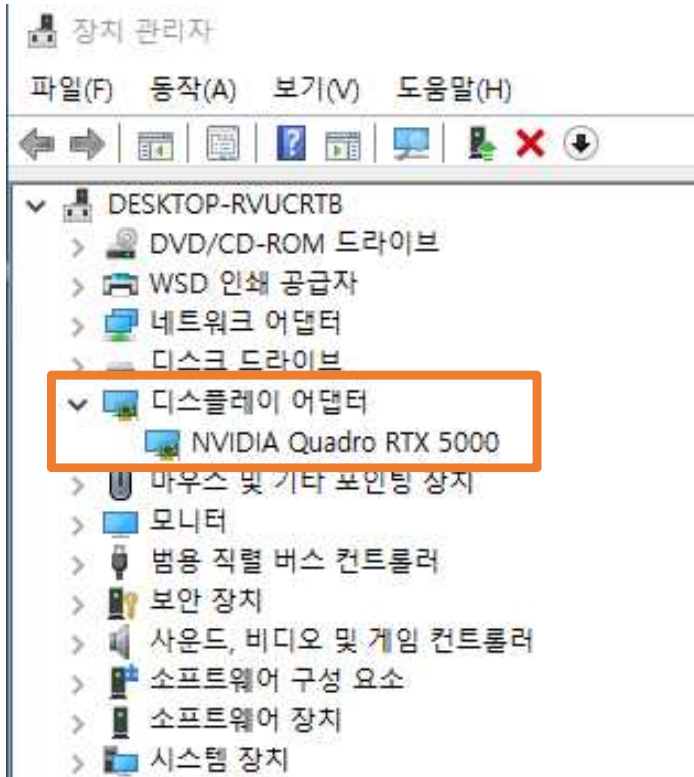


Pytorch GPU 버전 설치

CUDA 설치


- 그래픽 카드 확인
 - 윈도우 → 장치관리자 → 디스플레이 어댑터 → NVIDIA Quadro RTX 5000



CUDA 설치

- Computer Capability 확인
 - 하드웨어는 CUDA® Compute Capability 3.5 이상의 NVIDIA® GPU 카드만 지원
 - <https://developer.nvidia.com/cuda-gpus#compute> ← 사이트에서 3.5 이상인지 확인

If you have an older NVIDIA GPU you may find it listed on our [legacy CUDA GPU list](#).
Click the sections below to expand



CUDA-Enabled Quadro Products

Quadro Desktop Products

GPU	Compute Capability
Quadro RTX 8000	7.5
Quadro RTX 6000	7.5
Quadro RTX 5000	7.5
Quadro RTX 4000	7.5

CUDA 설치

- 최신 드라이버 설치 (<https://www.nvidia.com/Download/index.aspx?lang=kr>)

NVIDIA 드라이버 다운로드

아래의 드롭다운 목록에서 자신의 NVIDIA 제품에 알맞은 드라이버를 선택하세요. 1

제품 유형:

제품 시리즈:

제품 계열:

운영 체제:

다운로드 타입:

언어:

검색

NVIDIA RTX / QUADRO DESKTOP AND NOTEBOOK DRIVER RELEASE 460




버전: R460 U4 (461.72) WHQL
배포 날짜: 2021.2.26
운영 체제: Windows 10 64-bit
언어: Korean
파일 크기: 472.32 MB

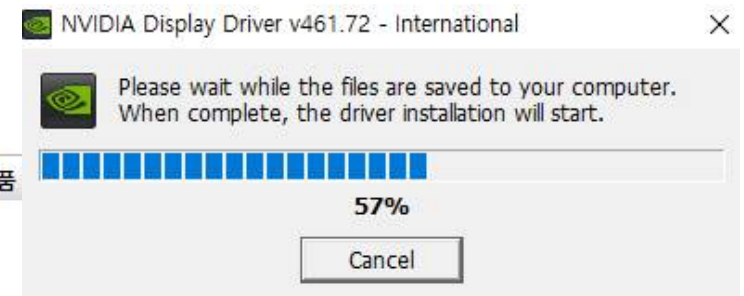
다운로드

사용자 가이드

제품

Driver Name Change

	461.72-quadro-rtx-desktop-notebook-w...	2021-03-16
	cuda_10.2.89_441.22_win10.exe	2021-03-16
	cudnn-10.2-windows10-x64-v8.1.0.77....	2021-03-16



CUDA 설치

- CUDA Toolkit 설치 <https://pytorch.org/get-started/locally/>
 - 본인의 환경에 맞는 설치파일 확인
 - CUDA Toolkit Archive 설치 페이지 (<https://developer.nvidia.com/cuda-toolkit-archive>)

START LOCALLY

Select your preferences and run the install command. Stable represents the most currently tested supported version of PyTorch. This should be suitable for many users. Preview is available if you want the latest, not fully tested and supported, 1.9 builds that are generated nightly. Please ensure that you **met the prerequisites below (e.g., numpy)**, depending on your package manager. Anaconda is recommended package manager since it installs all dependencies. You can also [install previous versions of PyTorch](#). Note that LibTorch is only available for C++.

Archived Releases

[CUDA Toolkit 11.2.1 \(Feb 2021\), Versioned Online Documentation](#)
[CUDA Toolkit 11.2.0 \(Dec 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.1.1 \(Oct 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.1.0 \(Sept 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.0 Update1 \(Aug 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.0 \(May 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 10.2 \(Nov 2019\), Versioned Online Documentation](#)
[CUDA Toolkit 10.1 update2 \(Aug 2019\), Versioned Online Documentation](#)
[CUDA Toolkit 10.1 update1 \(May 2019\), Versioned Online Documentation](#)
[CUDA Toolkit 10.1 \(Feb 2019\), Online Documentation](#)
[CUDA Toolkit 10.0 \(Sept 2018\), Online Documentation](#)

PyTorch Build	Stable (1.8.0)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 10.2	CUDA 11.1	ROCm 4.0 (beta)	None
Run this Command:	NOTE: Python 3.9 users will need to add '-c=conda-forge' for installation <code>conda install pytorch torchvision torchaudio cudatoolkit=10.2 -c pytorch</code>			

CUDA 설치

- CUDA Toolkit 설치 (다운로드 ➔ 설치)

CUDA Toolkit 10.2 Download


Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown.

Operating System	Windows	Linux	Mac OSX			
Architecture	x86_64					
Version	10	8.1	7	Server 2019	Server 2016	Server 2012 R2
Installer Type	exe (network)	exe (local)				

Download Installers for Windows 10 x86_64

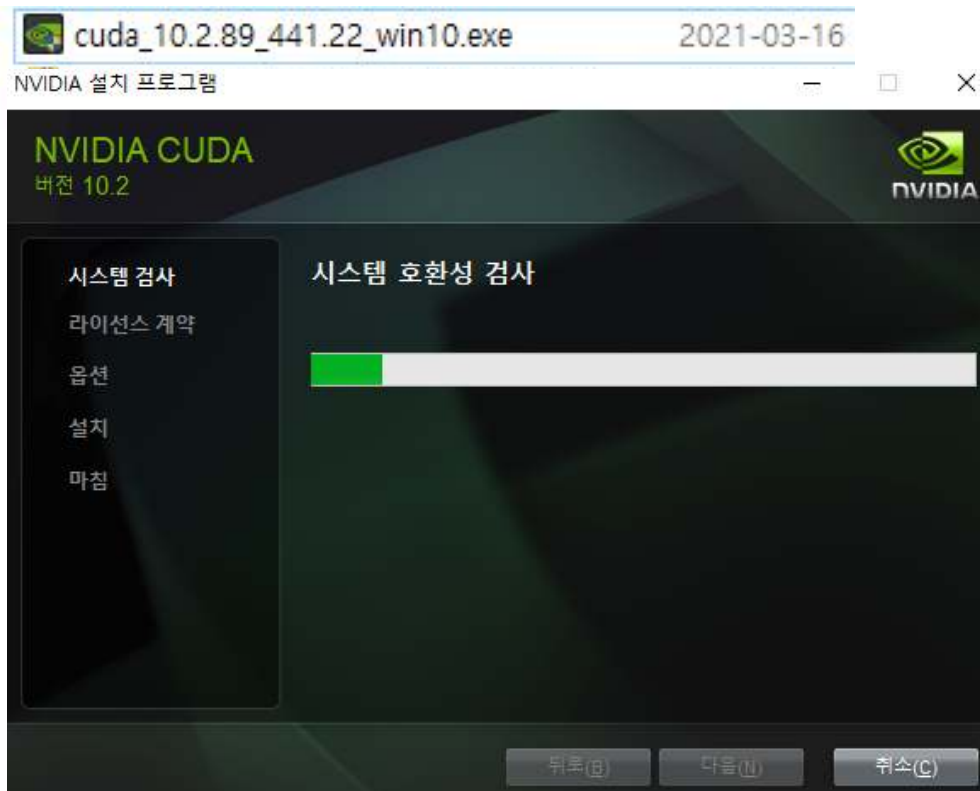
The base installer is available for download below.
There are 2 patches available. These patches require the base installer to be installed first.

> Base Installer	Download [2.6 GB] 
------------------	---

Installation Instructions:

CUDA 설치

- CUDA Toolkit 설치 (다운로드 ➔ 설치): 다음 다음 닫기.



CUDA 설치

- CuDNN 설치 (<https://developer.nvidia.com/cudnn>)
 - 회원가입 필요

NVIDIA cuDNN

The NVIDIA CUDA® Deep Neural Network library (cuDNN) is a GPU-accelerated library of primitives for deep learning routines such as forward and backward convolution, pooling, normalization, and more.

Deep learning researchers and framework developers worldwide rely on cuDNN to accelerate their software applications rather than spending time on low-level GPU programming. cuDNN is available for [Keras](#), [MATLAB](#), [MxNet](#), [PaddlePaddle](#), [PyTorch](#), and [TensorFlow](#). For access to the [NVIDIA GPU CLOUD](#) to learn more and get started.

Download cuDNN >

cuDNN Download Survey

cuDNN is a powerful library for Machine Learning. It has been developed to help developers learn and use the library. Please guide us on how you use this library by completing this short survey. You will be directed to the appropriate download page.

Start

What do you use cuDNN for?

- ☐ Training
- ☐ Inference
- ☒ Both

Do you currently use cuDNN via a framework (e.g. Tensorflow, PyTorch, others) or do you directly use cuDNN APIs?

- ☒ Use a framework such as Tensorflow, PyT, MxNet,...
- ☐ Directly use cuDNN APIs
- ☐ Both

What do you use cuDNN for?

- ☐ Training
- ☐ Inference
- ☒ Both

Do you currently use cuDNN via a framework (e.g Tensorflow, PyTorch, others) or do you directly use cuDNN APIs

- ☒ Use a framework such as Tensorflow, PyT, MxNet,...)
- ☐ Directly use cuDNN APIs
- ☐ Both

What network architectures closely resemble the ones you use today?

- ☒ Image Segmentation (UNet-3D, VNet)
- ☒ Object Detection (SSD, Mask R-CNN, Faster R-CNN)
- ☒ BERT, Transformer, GPT
- ☒ Speech Recognition / Synthesis (e.g Waveglow, WaveNet, RNN-T, Tacotron2, Jasper)

What framework(s) are you currently using to train deep learning models?

- ☐ TensorFlow
- ☒ PyTorch
- ☐ MXNet
- ☐ Other...(Please Specify)

What new features in cuDNN are you looking for?

- ☒ More samples and examples
- ☐ Enhanced coverage of 1D convolutions
- ☐ Flexible operator fusion support
- ☐ Enhanced coverage of grouped /depth-wise separable convolutions
- ☒ Enhanced coverage of 2D convolutions

cuDNN Download

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☒ I Agree To the Terms of the [cuDNN Software License Agreement](#)

Note: Please refer to the [Installation Guide](#) for release prerequisites, including supported GPU architectures and For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the [Deep Le](#)

[Download cuDNN v8.1.1 \(Feburary 26th, 2021\), for CUDA 11.0,11.1 and 11.2](#)

[Download cuDNN v8.1.1 \(Feburary 26th, 2021\), for CUDA 10.2](#)

[Archived cuDNN Releases](#)

will generate More variety deep learning model

CUDA 설치

[Download cuDNN v8.1.1 \(February 26th, 2021\), for CUDA 10.2](#)

Library for Linux, Ubuntu(x86_64)

cuDNN Library for Linux (x86)

cuDNN Library for Windows10 (x86)

cuDNN Runtime Library for Ubuntu18.04 (Deb)

cuDNN Developer Library for Ubuntu18.04 (Deb)

cuDNN Code Samples and User Guide for Ubuntu18.04 (Deb)

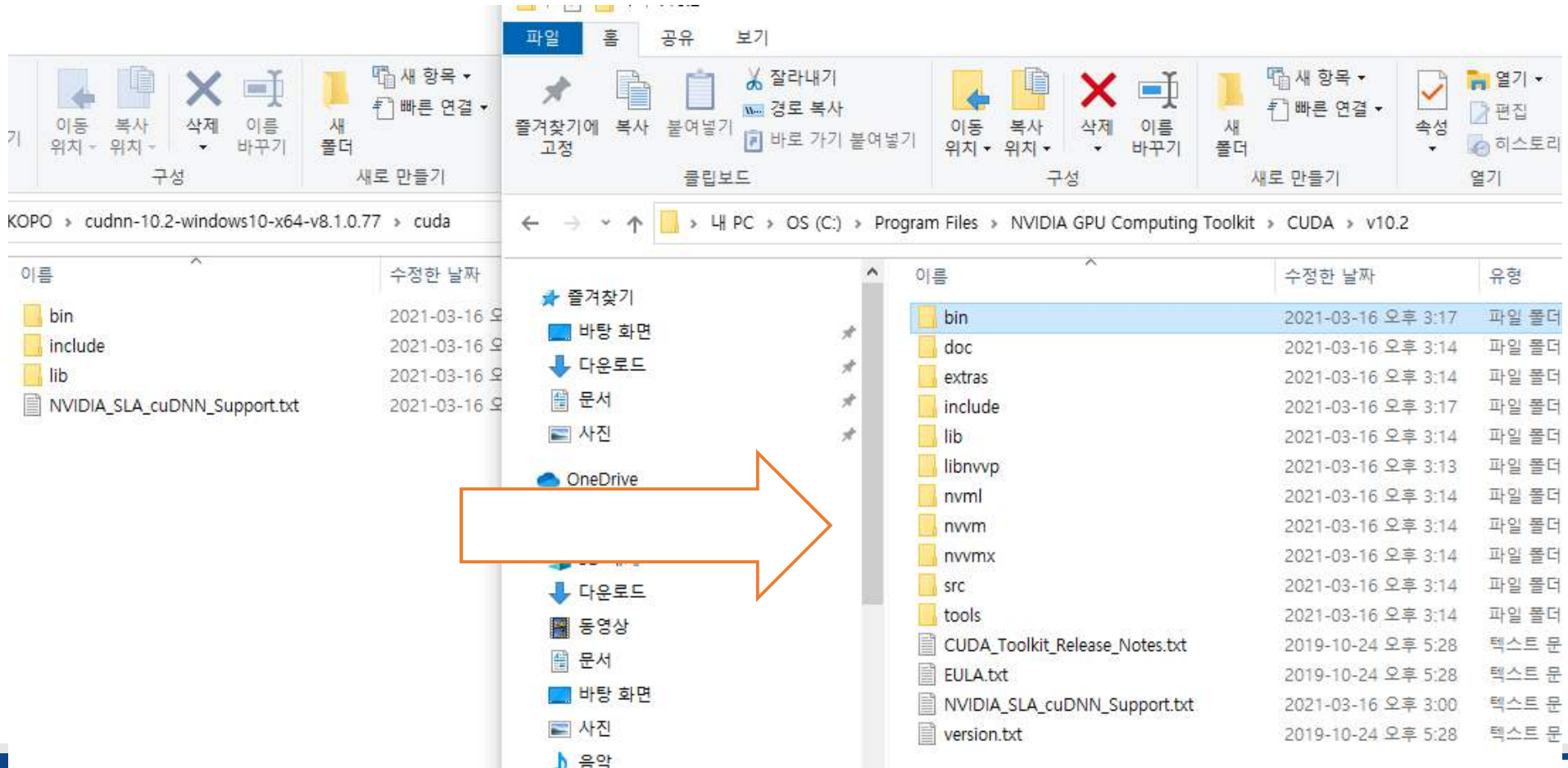
cuDNN Runtime Library for Ubuntu16.04 (Deb)

cuDNN Developer Library for Ubuntu16.04 (Deb)

cuDNN Code Samples and User Guide for Ubuntu16.04 (Deb)

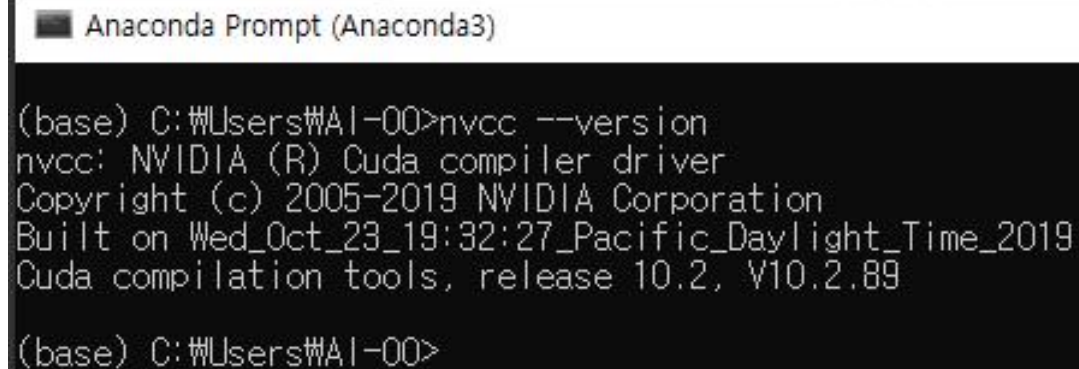
CUDA 설치

- CuDNN 압축 해제 후 CUDA Toolkit 폴더에 덮어쓰기.



CUDA 설치

- CUDA 설치 확인
 - Anaconda Prompt 실행
 - `nvcc --version`



```
Anaconda Prompt (Anaconda3)

(base) C:\Users\AI-00>nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Wed_Oct_23_19:32:27_Pacific_Daylight_Time_2019
Cuda compilation tools, release 10.2, V10.2.89

(base) C:\Users\AI-00>
```

Pytorch GPU 버전 설치하기

- 가상 환경 생성

- Anaconda prompt 실행

“ **conda create -n 가상환경이름 python anaconda** ” # 기본 라이브러리 함께 설치

```
(base) C:\Users\AI-00>conda create -n DL_Pytorch python=3.8 anaconda
```

```
done
#
# To activate this environment, use
#
#   $ conda activate DL_Pytorch
#
# To deactivate an active environment, use
#
#   $ conda deactivate
#
(base) C:\Users\AI-00>conda env list
# conda environments:
#
base                    * C:\ProgramData\Anaconda3
DL_Pytorch              C:\Users\AI-00\conda\envs\DL_Pytorch
```

Pytorch GPU 버전 설치하기

- 생성된 가상 환경에서 pytorch GPU 버전 설치
 - 생성한 가상환경 활성화: conda activate "가상환경명"

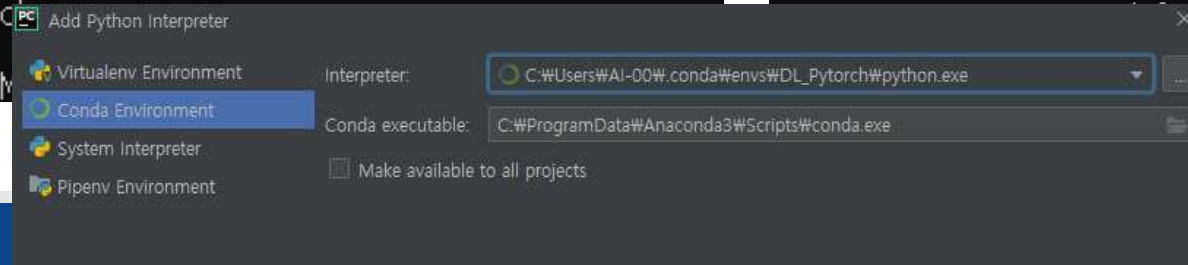
```
(base) C:\Users\AI-00>conda activate DL_Pytorch  
(DL_Pytorch) C:\Users\AI-00>
```

- Slide 5에 있는 CUDA Toolkit 버전에 맞는 Pytorch 설치

```
(DL_Pytorch) C:\Users\AI-00>conda install pytorch torchvision torchaudio cudatoolkit=10.2 -c pytorch  
Collecting package metadata (current_repodata.json): done  
Solving environment: done
```

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

```
Downloading and Extracting Packages  
torchvision-0.9.0 | 7.3 MB  
cudatoolkit-10.2.89 | 317.2 MB
```



conda list

```
watchdog 0.10.3 py38_0  
wcwidth 0.2.5 py_0  
webencodings 0.5.1 py38_1  
werkzeug 1.0.1 py_0  
wheel 0.35.1 py_0  
widgetsnbextension 3.5.1 py38_0  
win_inet_pton 1.1.0 py38_0  
win_unicode_console 0.5 py38_0  
wincertstore 0.2 py38_0  
winpty 0.4.3 4  
wrapt 1.11.2 py38he774522_0  
yamllint 0.2.5 py_0  
vapf 0.30.0 py_0  
xeromq 4.3.2 ha925a31_3  
xgboost 2.0.0 py_0  
xipp 3.4.0 pyhd3eb1b0_0  
xlib 1.2.11 h62dcd97_4  
py38_1  
py38_0  
py38he774522_0  
h04227a9_0
```

Pytorch GPU 버전 설치하기

- 설치 확인

The screenshot displays the PyCharm IDE interface. At the top, the 'Add Python Interpreter' dialog is open, showing the 'Conda Environment' selected. The 'Interpreter' field is set to 'C:\Users\AI-00\Anaconda3\envs\DL_Pytorch\python.exe', and the 'Conda executable' is 'C:\ProgramData\Anaconda3\Scripts\conda.exe'. Below the dialog, the 'pythonProject' is open in the editor, showing a file named 'test_pytorch_gpu.py'. The code in the file is as follows:

```
1 import torch
2 print(torch.cuda.get_device_name())
3 print(torch.cuda.is_available())
4 print(torch.__version__)
5
6 print("test")
7
```

To the right of the code editor, a terminal window is open, showing the output of the script:

```
(DL_Pytorch) C:\Users\AI-00>python
Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)] :: Anaconda3
Type "help", "copyright", "credits" or "license()" for more
>>> import torch
>>> torch.cuda.get_device_name()
'Quadro RTX 5000'
>>> torch.cuda.is_available()
True
>>> torch.__version__
'1.8.0'
>>>
```


Thank you

Q&A

www.kopo.ac.kr
jsshin7@kopo.ac.kr