

1. 에너지 데이터 분석환경 구축



에너지 데이터 분석 목적

■ 에너지 사용과 에너지 비용에 대한 이해

- 비즈니스 또는 조직 부문 및 자산을 기반으로 에너지 소비 및 비용을 분석할 수 있음
- 전기, 가스, 유류와 같은 유틸리티별 분석으로 사용량 관리방법을 전략화 할 수 있음

■ 에너지 사용량의 변동성을 확인하고 추세가 어떻게 변화하는지 이해

- 에너지 사용과 생산 간의 상관 관계를 보는 것과 같은 변수 간의 관계를 설정할 수 있음
- 에너지설비의 가장 적합한 작동모드를 결정하고 효율성을 높일 수 있는 기회를 식별할 수 있음

■ 목표 계산 및 진행상황 추적

- 에너지 사용/절감 목표를 계산하고 소비 수준에 대한 벤치마크 목표를 설정할 수 있음
- 에너지 소비를 줄이기 위해 구현한 새로운 전략의 성과를 추적할 수 있음

■ 에너지 수요 모델링

- 잠재적 에너지 수요를 모델링하고 다양한 "가정" 분석을 수행하는 데 사용
- 유틸리티 성능을 측정하고 순이익 관점에서 데이터 기반 프로그램을 실행할 수 있음

에너지 사용과 에너지 비용에 대한 이해

| 에너지 사용현황

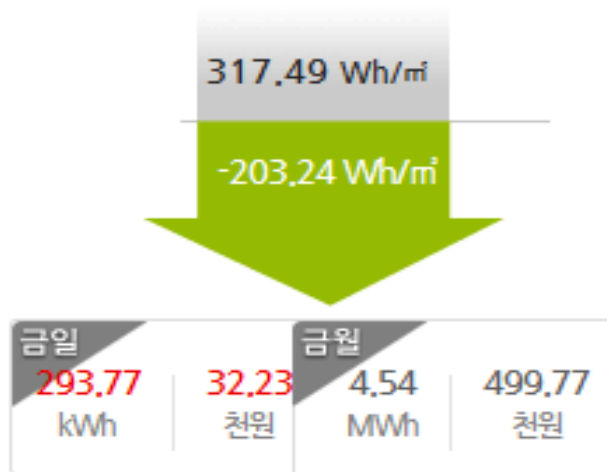
금일	금일	년간
34.54 MWh	3.67 백만원	586.72 MWh
	61.15 백만원	2.13 GWh
		219.8 백만원

| 에너지 지표

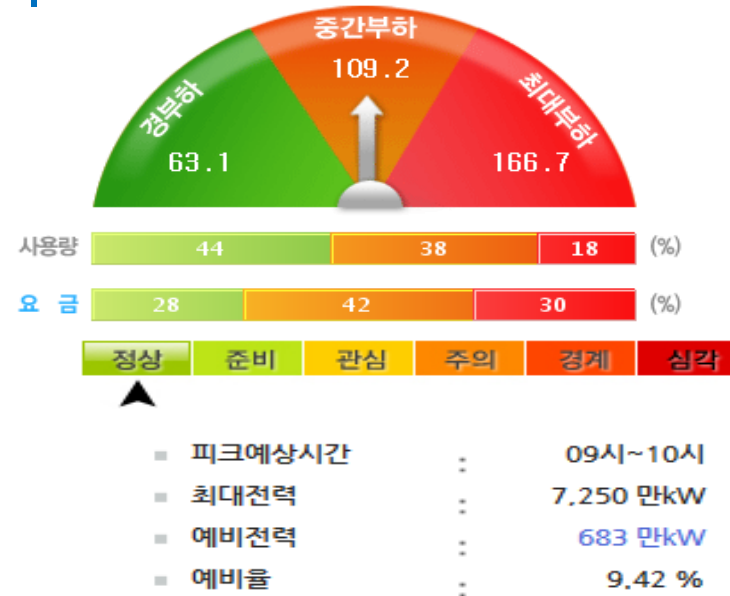
▶ 전일 에너지 지표현황(5종)

면적당소비량	114.25
면적당요금	13.75
인원당소비량	2,306.5
인원당요금	277.59
비근무일소비	45.85

▶ 전일 면적당소비량 평균 비교



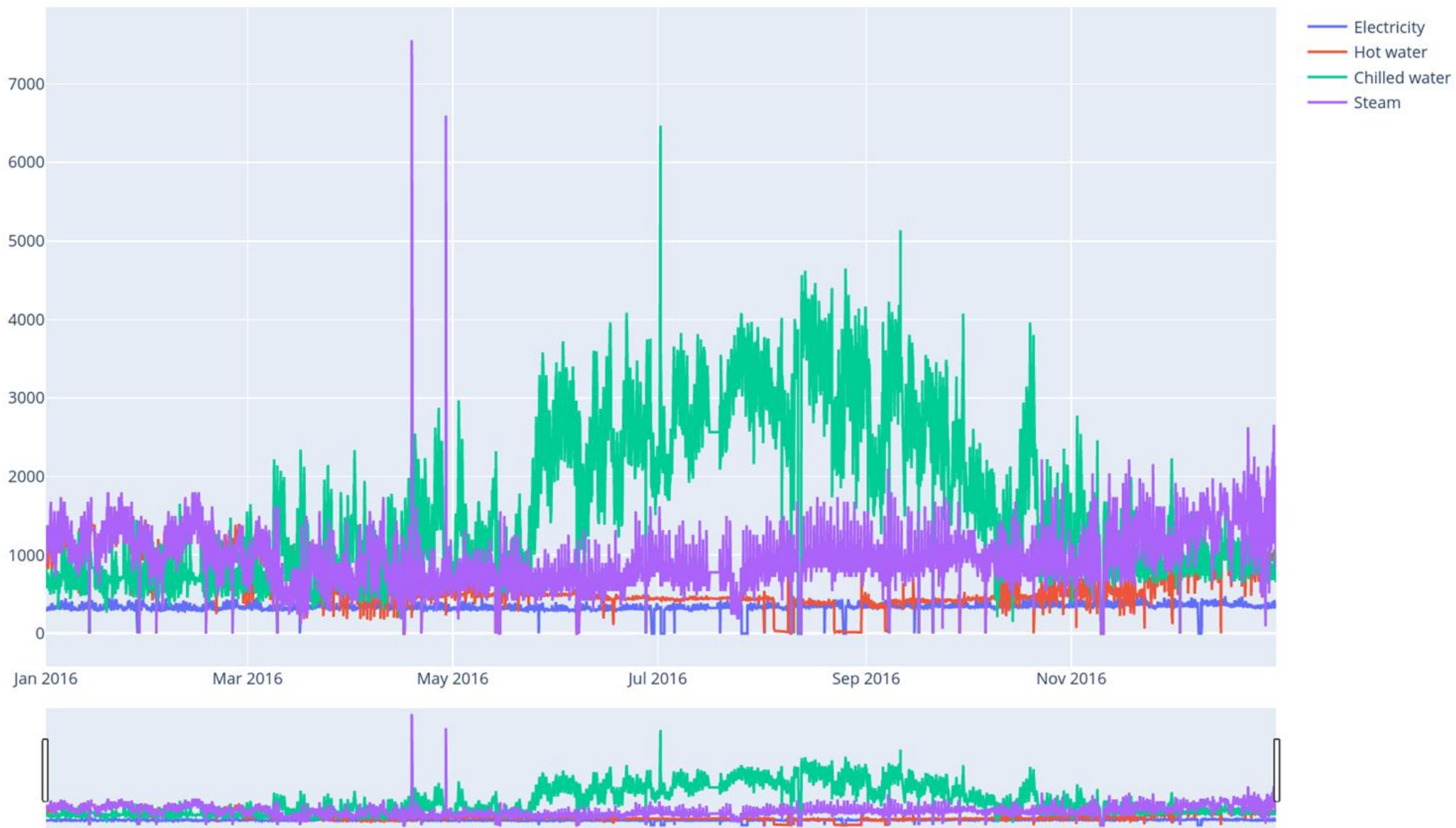
| 에너지 경보



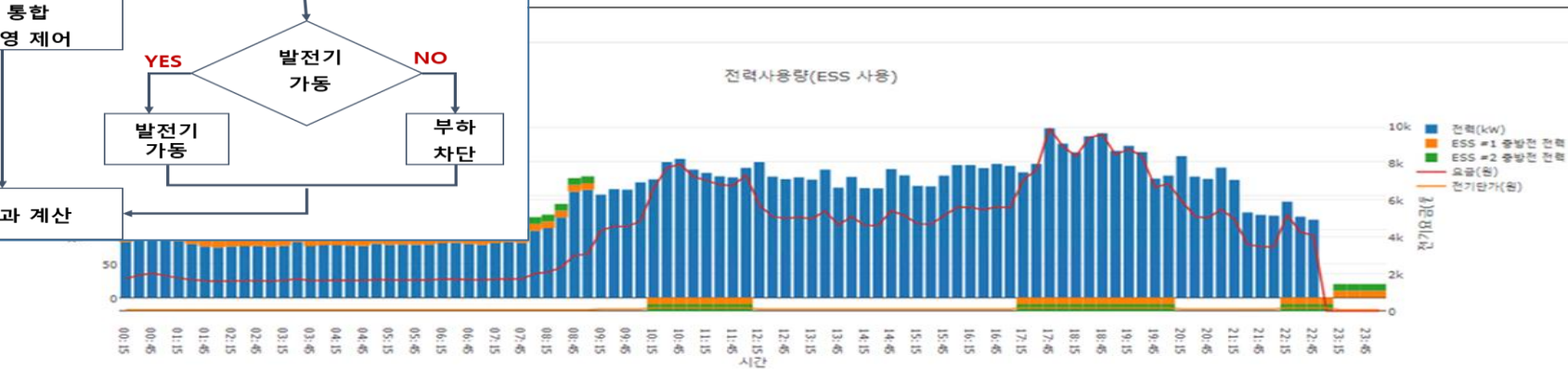
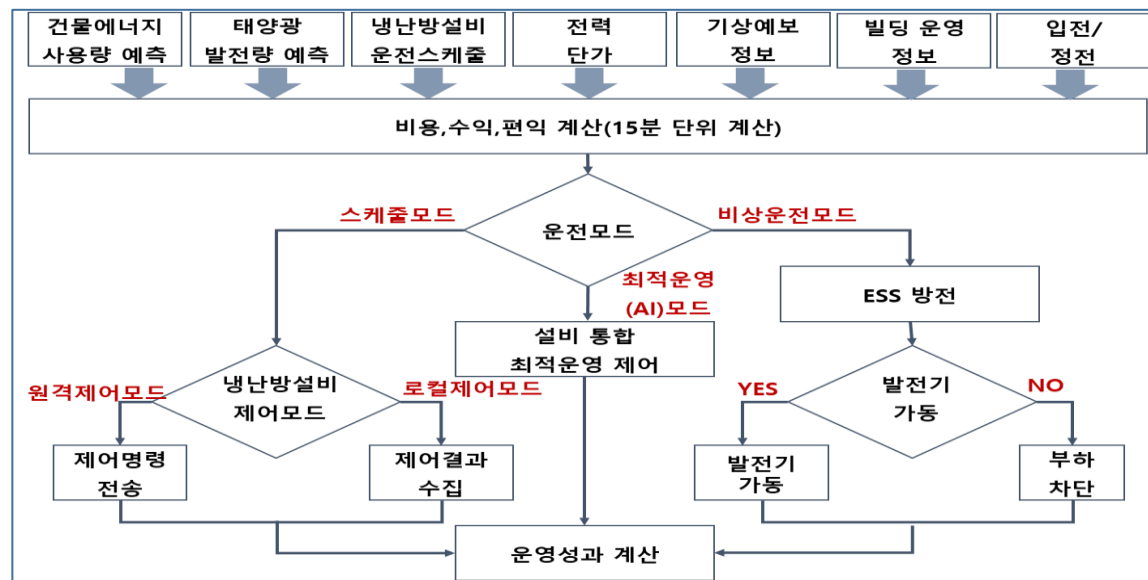
| 에너지 랭킹



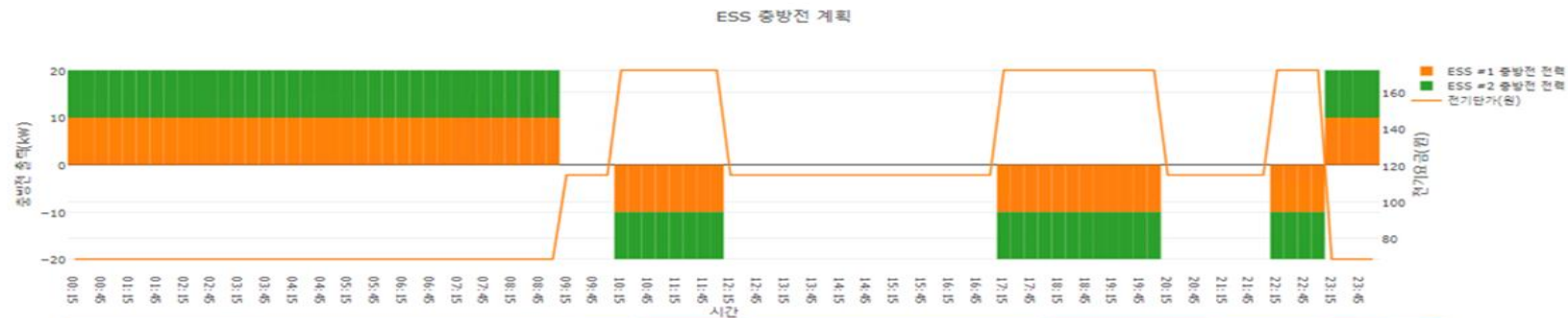
에너지 사용량 변동성 확인



목표 계산 및 진행상황 추적

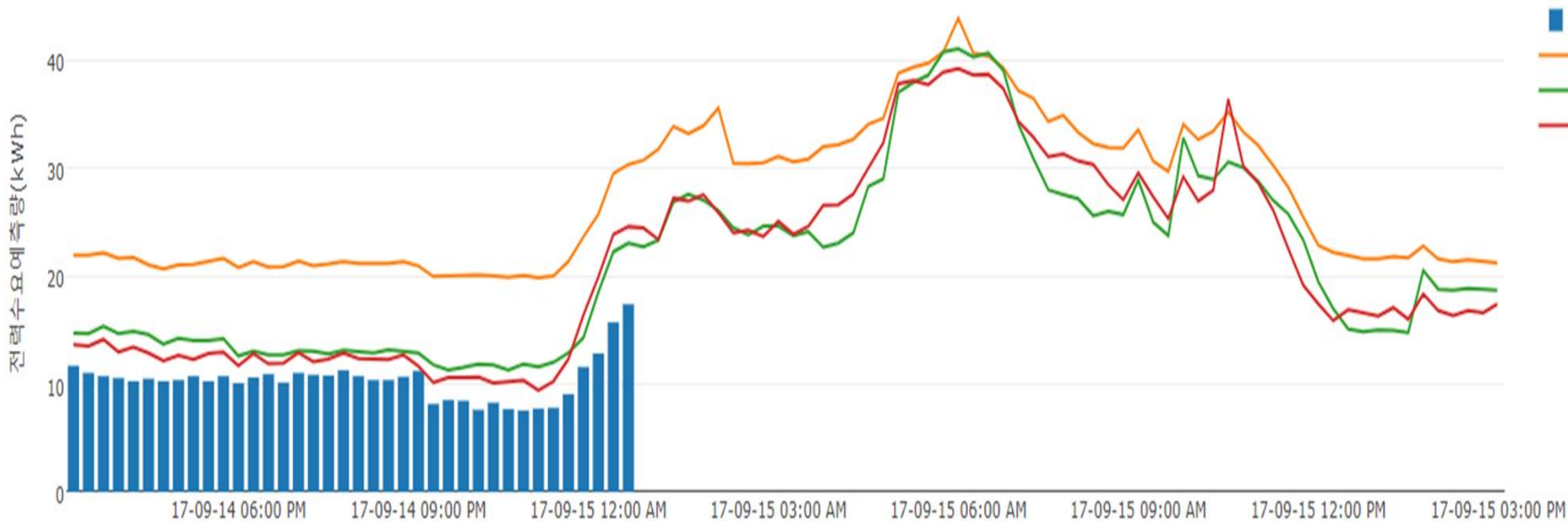


ESS 총방전계획



에너지 수요 모델링

전력 수요예측



실습환경

코드 에디터



Visual Studio Code



Sublime Text

PC



ANACONDA
Powered by Continuum Analytics®



Jupyter Notebook



Jupyter Lab

웹

colab

<https://colab.research.google.com/>

kaggle

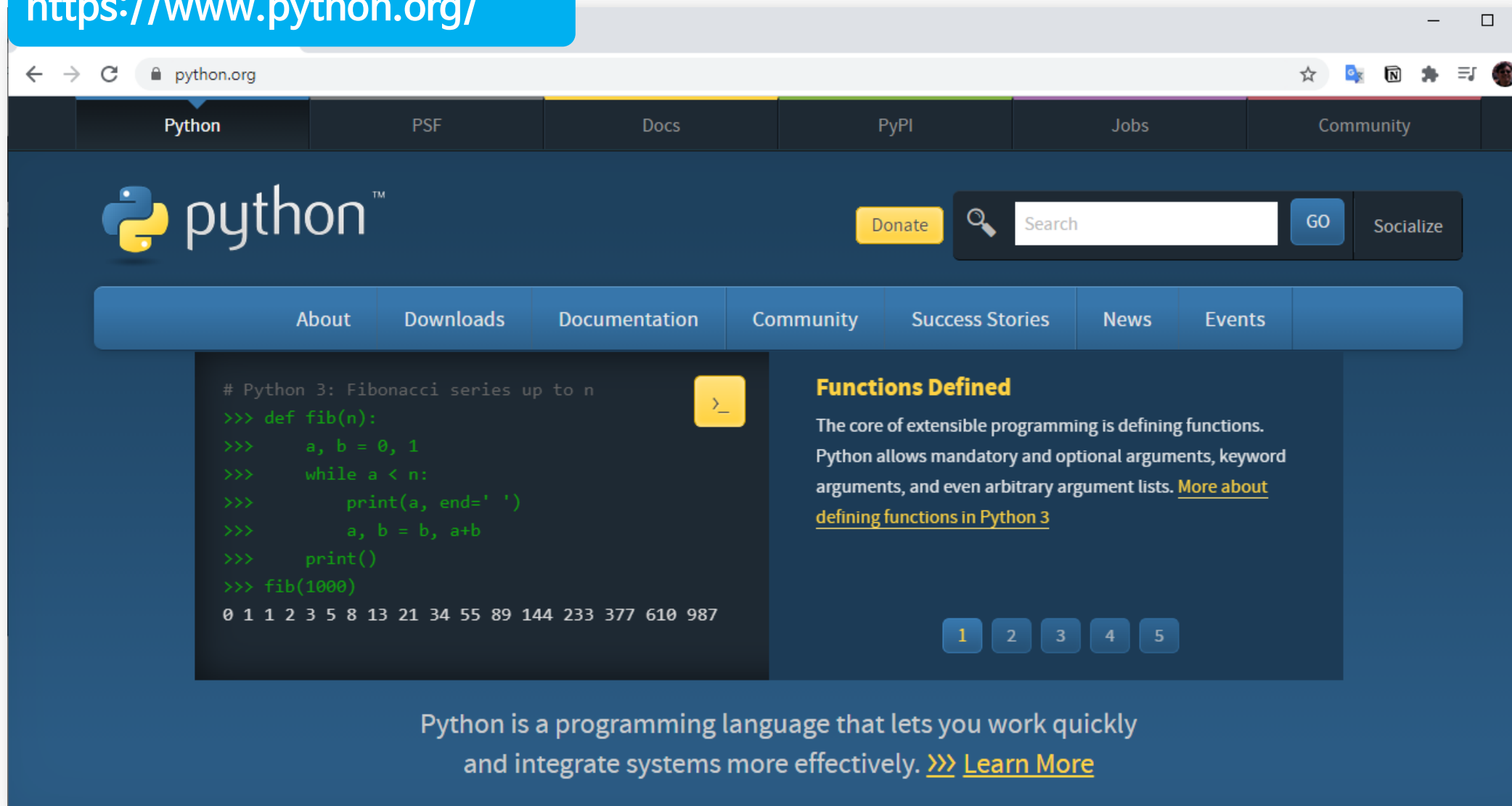
<https://www.kaggle.com/>

Benefits Of Python



파이썬 설치

https://www.python.org/



The screenshot shows the Python.org homepage. At the top, there's a navigation bar with links to Python, PSF, Docs, PyPI, Jobs, and Community. Below this is a dark blue header with the Python logo, a 'Donate' button, a search bar, and a 'Socialize' button. A secondary navigation bar contains links to About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a code editor on the left with a Python 3 Fibonacci script and its output, and a text block on the right titled 'Functions Defined' explaining the core of extensible programming. At the bottom, a blue banner states: 'Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)'.

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

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](#)

파이썬 설치

■ 파이썬 설치

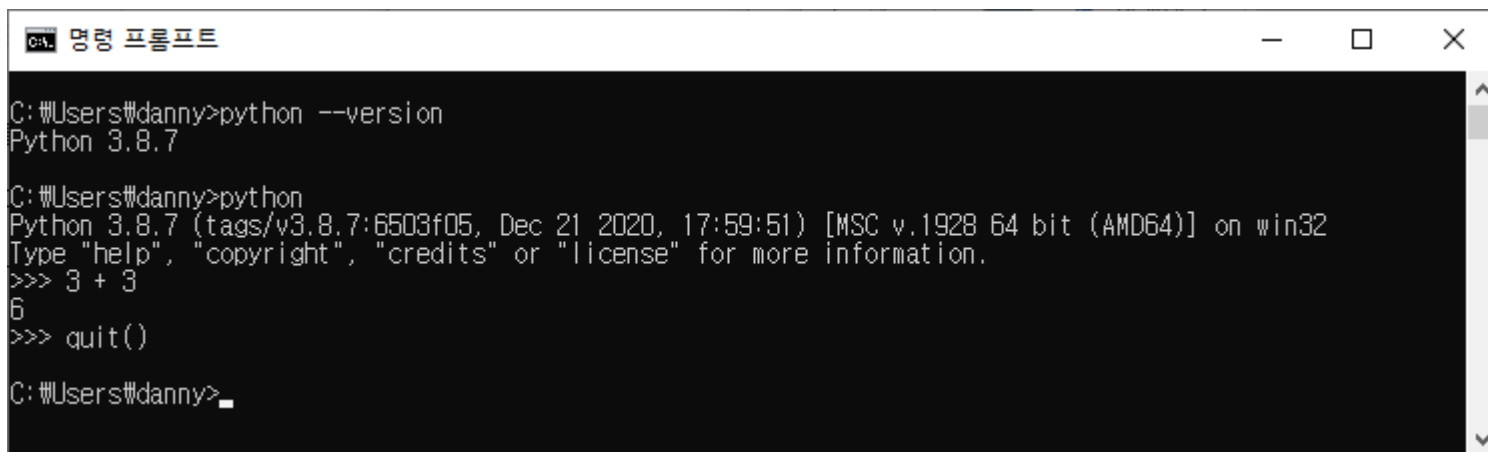
<https://www.python.org/downloads/>

Python 3.8.7

Release Date: Dec. 21, 2020

■ 파이썬 실행

- 버전 확인 : `python --version`
- 실행 : `python`
- 종료 : `quit()`



```
C:\Users\danny>python --version
Python 3.8.7

C:\Users\danny>python
Python 3.8.7 (tags/v3.8.7:6503f05, Dec 21 2020, 17:59:51) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 3 + 3
6
>>> quit()

C:\Users\danny>
```

파이썬 가상환경 설치

Virtual Environment 구성으로 프로젝트별로 독립된 파이썬 실행 환경을 사용할 수 있습니다.

■ 가상환경 생성

- `python -m venv venv`

■ 가상환경 실행

- Windows : `venv\Scripts\activate.bat`
- Linux, macOS : `source venv/bin/activate`

■ 패키지 설치

- `pip install pandas`
- `pip install tensorflow`
- `pip install jupyter`

■ 패키지 목록 관리

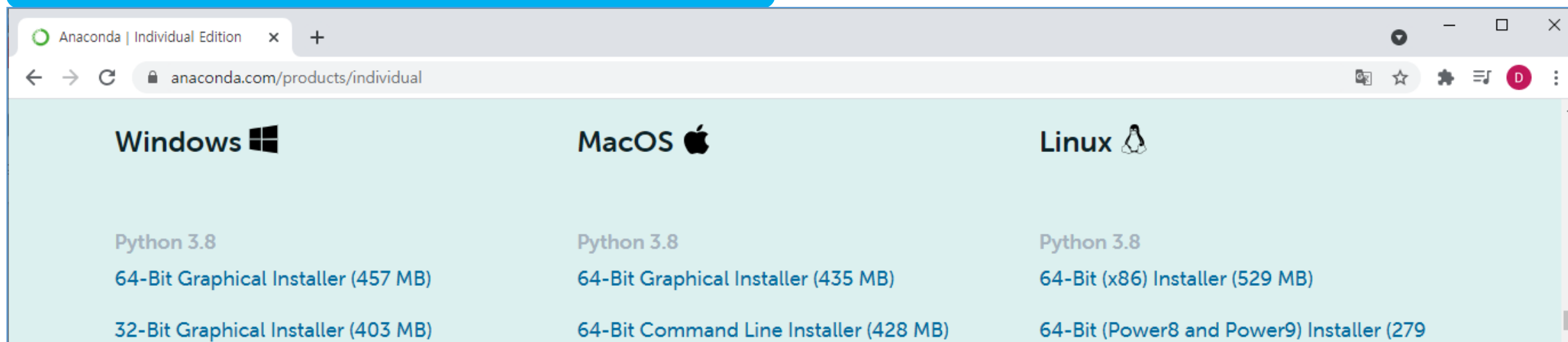
- `pip freeze > requirements.txt`
- `pip install -r requirements.txt`

아나콘다(Anaconda) 설치

Anaconda는 데이터 분석에 필요한 패키지들과 통합 개발환경을 포함한 배포판입니다.

■ 파이썬 설치/실행

<https://www.anaconda.com/download/>



■ 파이썬 실행

- 윈도우 시작메뉴 -> Anaconda3 -> Anaconda Prompt -> python

■ 가상환경

- conda create --name myenv
- conda activate myenv

아나콘다(Anaconda)

The screenshot shows the Anaconda Navigator desktop application. The window title is "Anaconda Navigator". The top menu bar includes "File" and "Help". The top right corner has an "Upgrade Now" button and a "Sign in to Anaconda.org" button. The left sidebar contains navigation links: "Home", "Environments", "Learning", and "Community". The main area displays a grid of application tiles for the "base (root)" environment. Each tile includes an icon, the application name, version, a brief description, and a "Launch" button. The applications shown are:



- CMD.exe Prompt** (0.1.1): Run a cmd.exe terminal with your current environment from Navigator activated.
- JupyterLab** (2.1.5): An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
- Notebook** (6.0.3): Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
- Powershell Prompt** (0.0.1): Run a Powershell terminal with your current environment from Navigator activated.
- PyCharm** (2021.1.1): Full-Featured Python IDE by JetBrains. Supports code completion, linting, debugging, and domain-specific enhancements for web development and data science.
- Qt Console** (4.7.5): PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.
- Spyder** (4.1.4): Scientific PYTHON Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features.
- VS Code** (1.55.2): Streamlined code editor with support for development operations like debugging, task running and version control.

At the bottom left of the sidebar is a "Documentation" link.

실습환경 - Chrome

컴퓨터에 Chrome이 설치되어 있지 않은 경우, Chrome을 다운로드하여 설치 하고 기본 웹 브라우저를 Chrome로 설정하세요.

Windows 10

1. 컴퓨터에서 시작 메뉴 를 클릭합니다.
2. 설정 을 클릭합니다.
3. 기본 앱을 엽니다.
 - 기존 버전: 시스템 > 기본 앱을 클릭합니다.
 - 크리에이터스 업데이트: 앱 > 기본 앱
4. 하단의 '웹 브라우저'에서 현재 브라우저를 클릭합니다. 일반적으로 Microsoft Edge입니다.
5. '앱 선택' 창에서 **Chrome**을 클릭합니다.

나중에 손쉽게 Chrome을 열려면 작업 표시줄에 단축키를 추가하세요.

1. 컴퓨터에서 Chrome을 엽니다.
2. 하단의 Windows 작업 표시줄에서 Chrome을 마우스 오른쪽 버튼으로 클릭합니다.
3. 작업 표시줄에 고정을 클릭합니다.

출처 : <https://bit.ly/30DvgKY>

실습환경 - 구글 코랩(Colab)

개발툴 설치없이 웹상에서 파이썬 프로그램을 할수 있는 환경으로 딥러닝에 필요한 GPU를 사용할 수 있습니다.

<https://colab.research.google.com> **구글 계정 필요**



The screenshot displays the Google Colaboratory web interface. At the top, a blue banner shows the URL <https://colab.research.google.com> and a red button indicating '구글 계정 필요' (Google account required). Below the banner, the interface includes a left sidebar with navigation options like 'Colaboratory 소개' (Colaboratory Introduction), '시작하기' (Get started), '추가 리소스' (Additional resources), and '머신러닝 예제: Seedbank' (Machine learning examples: Seedbank). The main area shows a 'Welcome to Colaboratory' message. A 'Runtime' menu is open, listing various execution options with keyboard shortcuts. A 'Note Settings' dialog box is also visible, showing the 'Hardware accelerator' dropdown set to 'GPU'.

Colaboratory에 오신 것을 환영합니다

파일 수정 보기 삽입 런타임 도구 도움말

+ 코드 + 텍스트 + 드라이브

목차 코드 스니펫

Colaboratory 소개

시작하기

추가 리소스

머신러닝 예제: Seedbank

섹션

모두 실행 Ctrl+F9

이전 셀 실행 Ctrl+F8

초점이 맞춰진 셀 실행 Ctrl+Enter

선택항목 실행 Ctrl+Shift+Enter

이후 셀 실행 Ctrl+F10

실행 중단 Ctrl+M

런타임 다시 시작 Ctrl+M

다시 시작 및 모두 실행

런타임 초기화

런타임 유형 변경

세션 관리

런타임 로그 보기

노트 설정

하드웨어 가속기 GPU

Colab를 최대한 활용하려면 필요하지 않은 경우 GPU를 사용하지 않는 것이 좋습니다.

[자세히 알아보기](#)

☐ 이 노트를 저장할 때 코드 셀 출력 생략

취소 저장

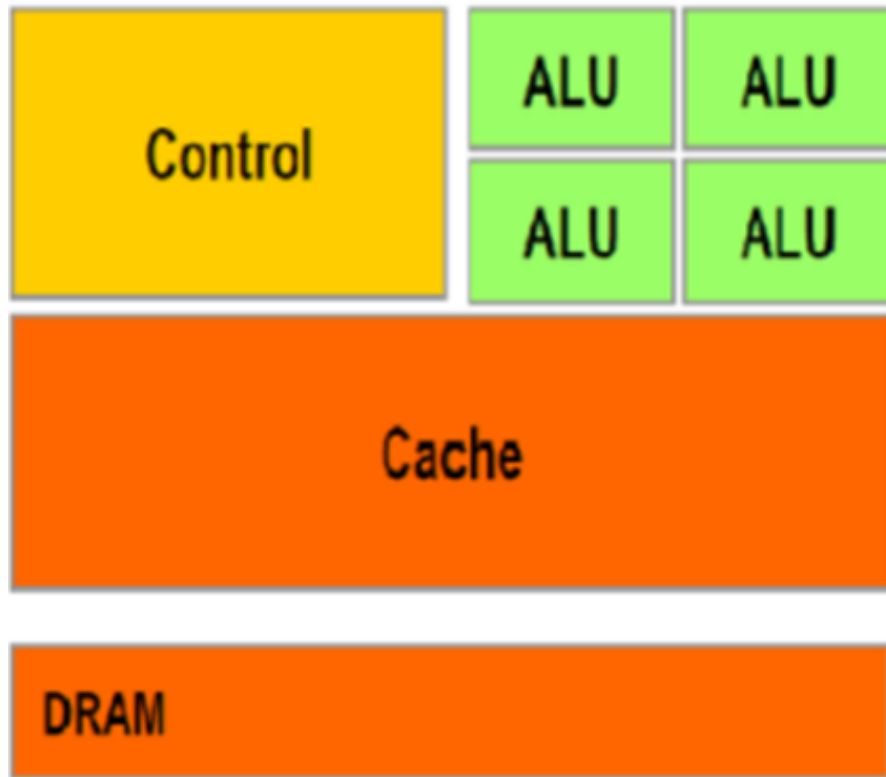


고성능GPU(Graphics Processing Unit)

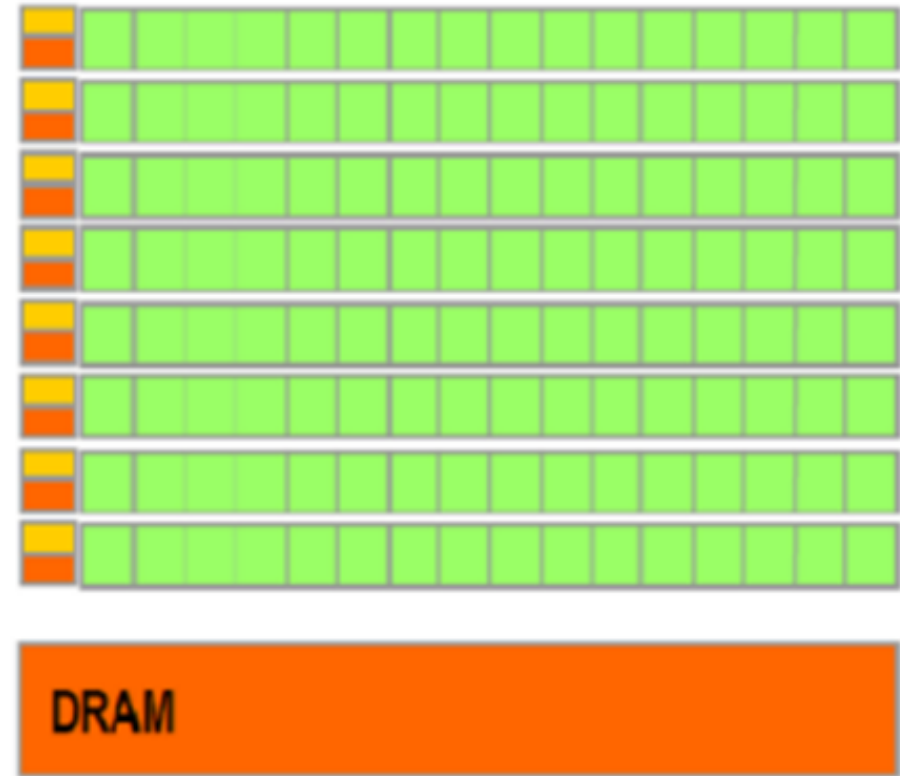
CPU vs GPU

CPU는 제어와 복잡한 연산을 수행하는 구조이며,
GPU는 단순 연산 특화 구조로 병렬로 고속연산처리를 할 수 있습니다.

■ CPU 구조



■ GPU 구조

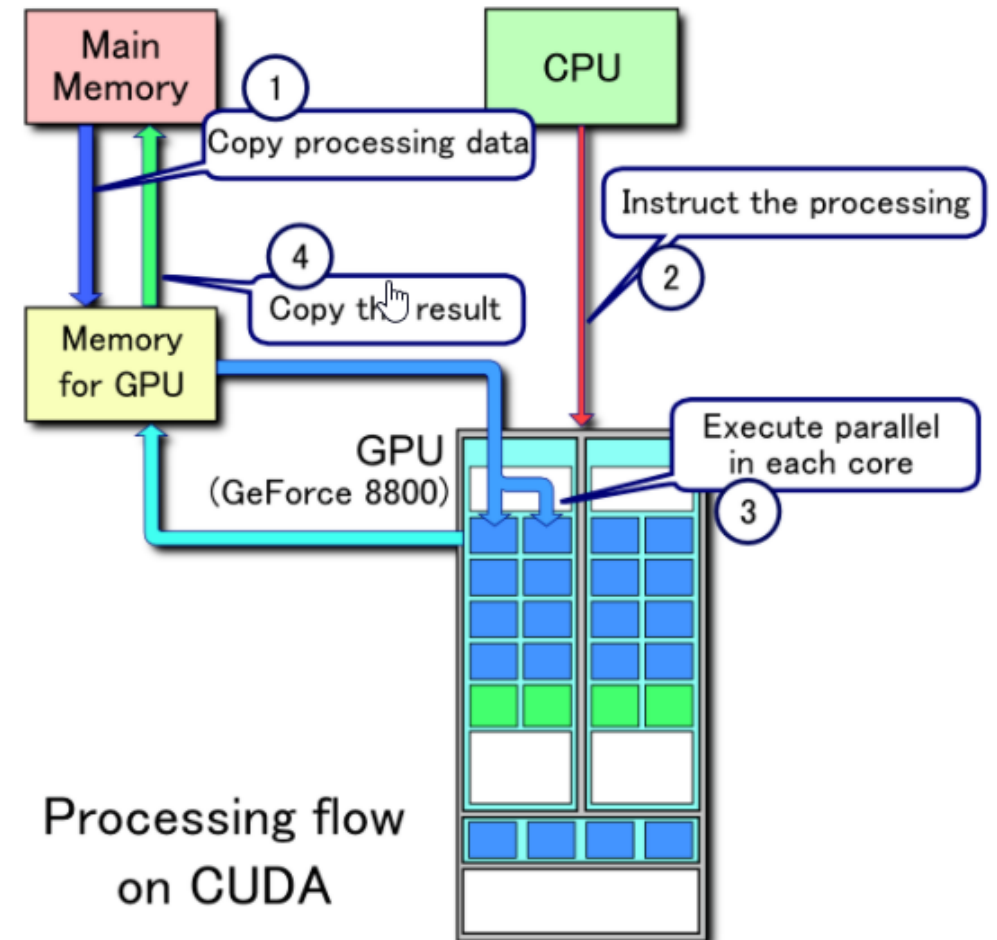
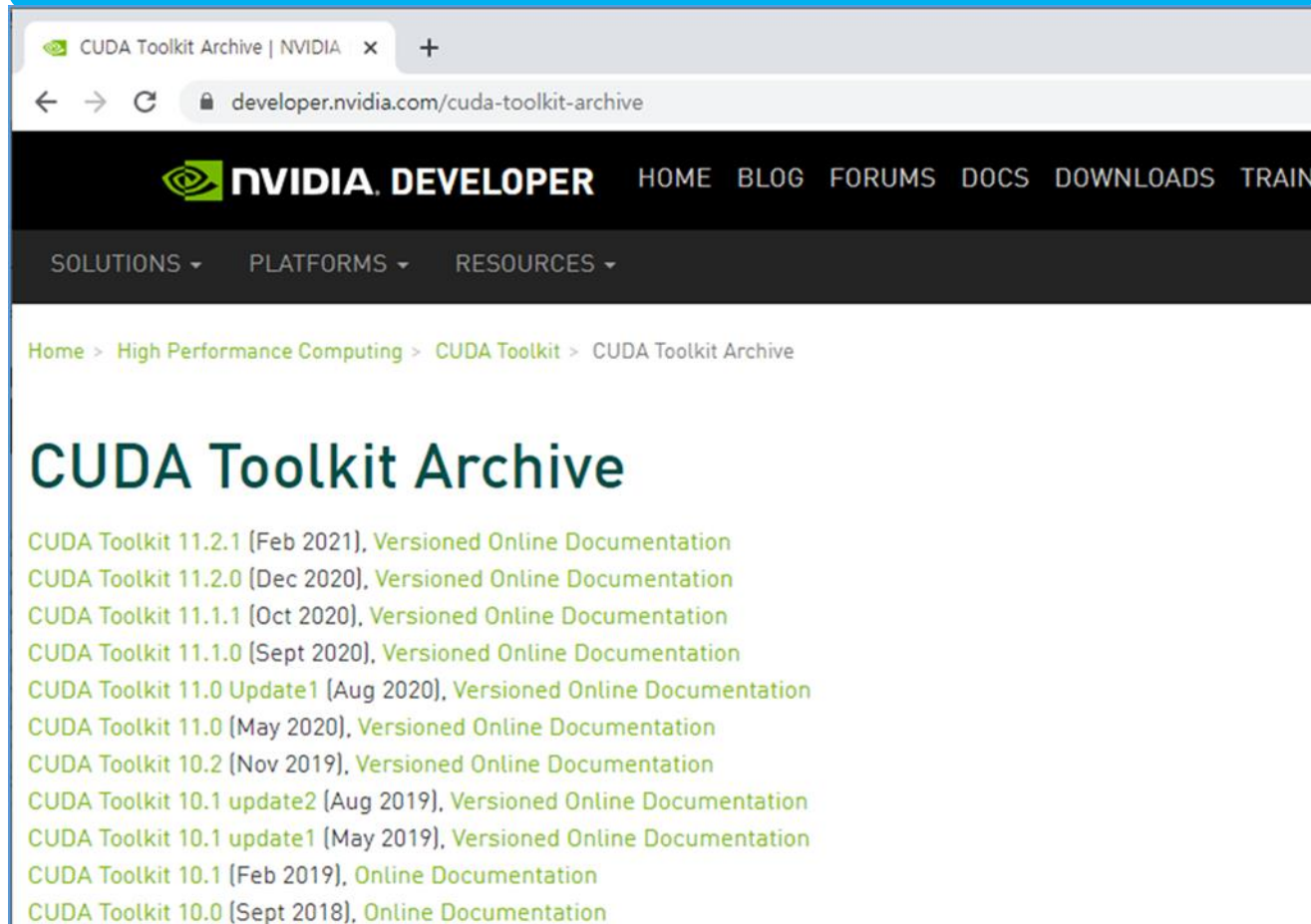


ALU(Arithmetic Logic Unit) : 산술연산장치

CUDA(Compute Unified Device Architecture) 설치

CUDA는 GPU에서 병렬처리를 수행하는 알고리즘을 사용할 수 있도록 해주는 GPU Computing SDK입니다.

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

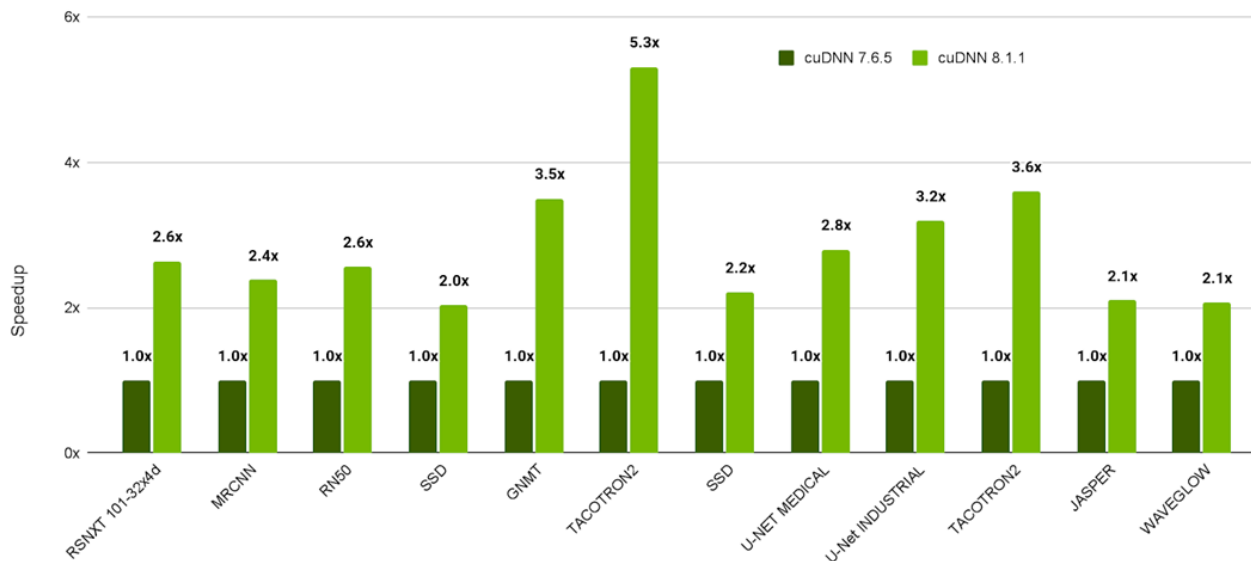


CuDNN(CUDA Deep Neural Network Library) 설치

DNN(Convolution, Pooling, Normalization, Activation) 루틴을 빠르게 수행하는 GPU 가속화 라이브러리입니다.

<https://developer.nvidia.com/cudnn>

A100 OVER 5X FASTER THAN V100 WITH CUDNN 8.1



cuDNN Accelerated Frameworks



참고 : <https://n1094.tistory.com/42>
<https://blogs.nvidia.co.kr/2018/01/16/cuda-toolkit/>

파이썬 주요 패키지(라이브러리)



행렬과 다차원 배열을 쉽게 처리 할 수 있게 해주는 라이브러리

pandas

데이터를 처리하고 분석하는 데 효과적인 패키지



데이터를 차트나 플롯(Plot)으로 그려주는 시각화 패키지



matplotlib을 기반으로 다양한 색상 테마와 통계용 차트 등의 기능을 추가한 시각화 패키지



교육 및 실무를 위한 머신러닝 패키지



TensorFlow 구글에서 만든 오픈소스 딥러닝 프레임워크



페이스북에서 만든 오픈소스 딥러닝 프레임워크

파이썬 기초

■ 변수 할당(Variable Assignment)

```
x = 2
y = 3
z = x + y
```

```
x = 'hello'
```

Single Quotation
작은 따옴표

```
x = "hello"
```

Double Quotation
쌍 따옴표

```
x
```

```
[Out] 'hello'
```

■ 출력

```
print(x)
```

```
[Out] 'hello'
```

■ 리스트(List)

```
[1, 2, 3]
```

```
['a', 'b', 'c']
```

```
my_list = [1, 2, 'apple', True]
```

Bracket
대괄호

```
my_list.append(100)
```

```
my_list[0]
```

```
my_list[:-1]
```

```
my_list[-1]
```

■ 딕셔너리(Dictionary)

```
d = {'key1': 'item1', 'key2': 'item2'}
```

Brace
중괄호

```
d['key1']
```

```
[Out] 'item1'
```


파이썬 실습



`python_essence.ipynb`

`python_tutorial.ipynb`

Thank you