

AI활용 표현과 문제해결

[Practice Settings & Numpy]

허세훈

충남대학교
지능소프트웨어연구실
2021.03.08



Sehun Hu (허세훈)

E-mail : sehunhu5247@gmail.com

Laboratory : Intelligent Software

Office : W5602

실습 수업 진행방향

* 기본적으로 Python과 선형대수를 배우고 왔음을 가정

매주 실시간 강의

YouTube 링크를 통해 실습 수업시간에 진행
→ 출석은 성적에 반영

YouTube 댓글을 통해
출석 확인

수업 시작 시점부터 1시간 이내에
실시간 채팅에 자신의 학번과 이름을 입력

이론 시간의 내용 보완

이론 시간에 배웠던 내용들을
실제로 구현해보는 시간으로 활용 예정

과제 x
기말고사 반영 o

매주 실습에 따른 과제는 없으나,
기말고사 때 실습 시간에 배운 내용들이
나올 수 있음

이번 주차 실습 내용

Pycharm 설치

- 학생 계정 생성
- Pycharm Professional 설치

Anaconda

- 소개
- 설치
- 가상환경

Jupyter Notebook

- 소개
- 실행 및 예시

Numpy

- 소개
- 설치
- 배열/연산

PyCharm 설치

1. 학교 포털의 “학생 웹메일(Office 365)” 메뉴를 통해 계정 만들기

Office 365

충남대학교
CHUNGNAM NATIONAL UNIVERSITY

Microsoft Teams 도움말 센터

도움말

HOME

가입하기

안내

이용약관 | 정보입력 | 가입완료

이용약관

☒ 이용약관/개인정보수집 및 이용 모두 동의

제 1 조. 목적

본 약관은 주식회사 (주)프리스티앤씨(이하 "회사"라고 함)의 인터넷 사이트, '대학 Office 365 통합 가입 센터' 사이트 (<http://www.o365univ.net>) 이하 "사이트"라 함)를 통하여 제공하는 인터넷 관련 서비스, Microsoft Office 365 설치지원 서비스 및 기타 정보 서비스(이하 "서비스"라고 함)와 관련하여 회사와 고객간의 권리와 의무, 책임사항 및 회원의 서비스 이용절차에 관한 사항을 규정함을 목적으로 합니다.

제 2 조. 약관의 효력과 변경

1. 이 약관은 인터넷 서비스 이용약관의 성격상 법 제24조 제2항에 의거하여 웹사이트인 대학 Office 365 통합 가입 센터 홈페이지 (<http://www.o365univ.net>)에 온라인으로 이 약관을 명시하고, 고객이 약관에 대해 "동의" 버튼을 클릭함으로써 효력이 발생합니다.

2. 회사는 전자상거래 등에서의 소비자보호에 관한 법률, 약관의 규제에 관한 법률, 전자거래 기본법, 전자서명법, 정보통신망 이용촉진 등에 관한 법률,

☒ 서비스 이용약관에 동의합니다.(필수)

'(주)프리스티앤씨'(이하 "회사")는 고객님의 개인정보를 중요시하며, "정보통신망 이용촉진 및 정보보호"에 관한 법률을 준수하고 있습니다. 회사는 개인정보취급방침을 통하여 고객님께서 제공하시는 개인정보가 어떠한 용도와 방식으로 이용되고 있으며, 개인정보처리방침을 개정하는 경우 웹사이트를 통하여 공지할 것입니다.

1. 수집하는 개인정보 항목 및 수집방법

회사는 회원가입, 상담, 서비스 신청 등의 서비스 제공을 위해 최소한의 개인정보를 수집하고 있습니다.

- 수집항목

[필수항목] 아이디, 비밀번호, 휴대폰번호, 소속학교, E-Mail, 표시이름

- 개인정보 수집방법 : 홈페이지(회원가입), 상담 및 문의(전화, 홈페이지), 설문 및 이벤트 참여, 제휴사의 제공 등

- 다. 비밀번호는 최초 회원가입 시 Office 365 메일을 위해서 필수하나 자체 보관은 하지 않습니다.

☒ 개인정보 취급방침에 동의합니다.(필수)

Office 365 서비스 제공과 관련하여 국외서버를 이용함에 따라 다음과 같은 내용을 공지하며, 사용자 개인정보에 대한 국외이전동의를 구하고자 합니다. 이전된 데이터는 개인정보취급방침을 준수하며, 회원의 동의가 완료된 이후에는 서비스 발효이전까지 Microsoft가 제공하는 Office 365 서비스 사용을 위한 개인정보 국외이전을 동의하는 것으로 간주합니다

1. 이전항목

- 사용 계정 (학번 또는 고유 ID)

- 이름 (성명)

- 휴대폰/개인 이메일(본실 시 본인 확인 수단)

- 교육기관용 Office 365 및 설치할 오피스 (Office 365 ProPlus) 라이선스

2. 이전 국가, 아시아, 파병

☒ 개인정보 국외이전에 동의합니다.(필수)

동의

비동의

HOME

가입하기

안내

이용약관 | 정보입력 | 가입완료

정보입력

학교

충남대학교

로그인 ID

아이디 입력

@o.cnu.ac.kr

중복확인

이름

이름 입력

비밀번호

비밀번호 입력

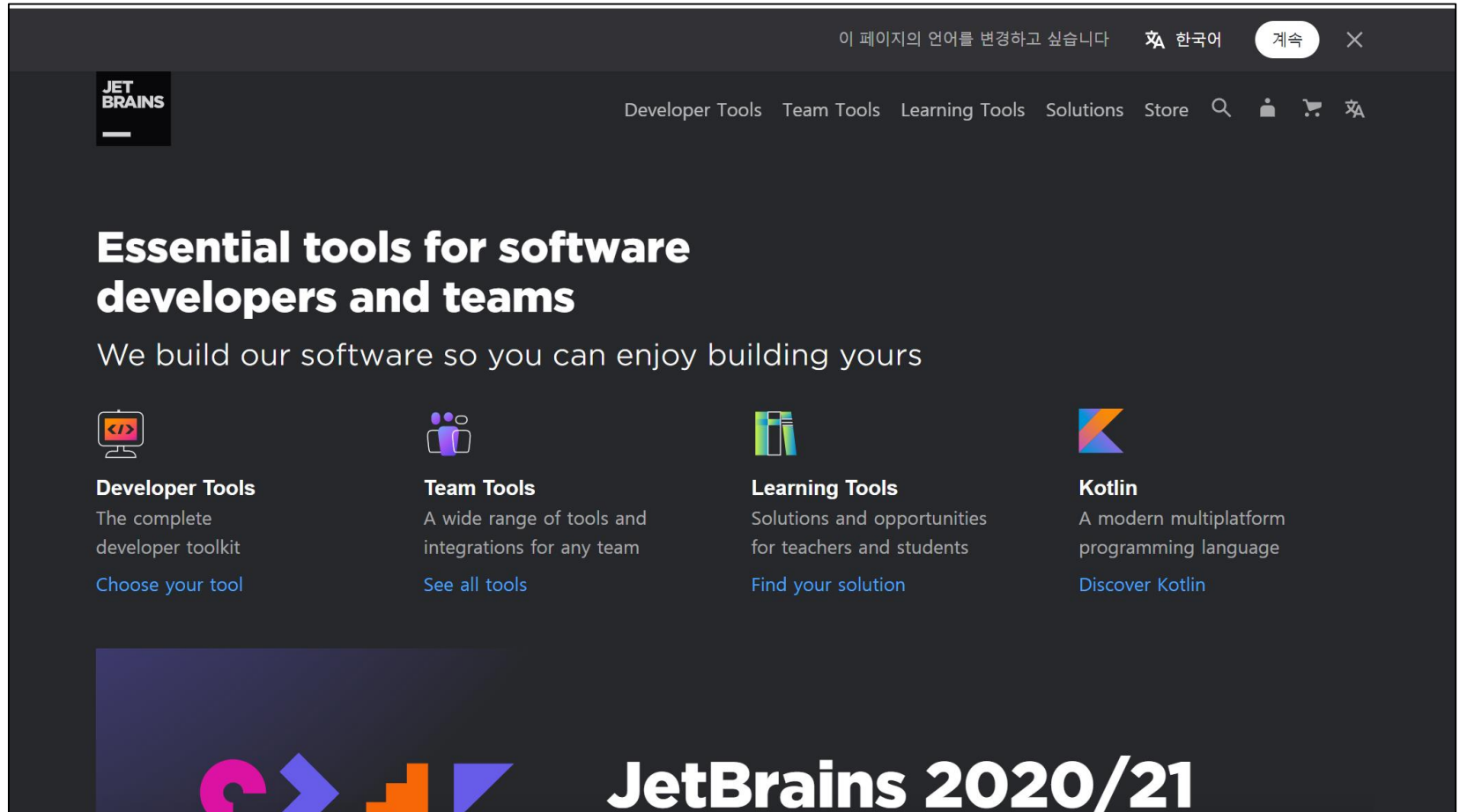
비밀번호 확인

비밀번호 입력

가입하기

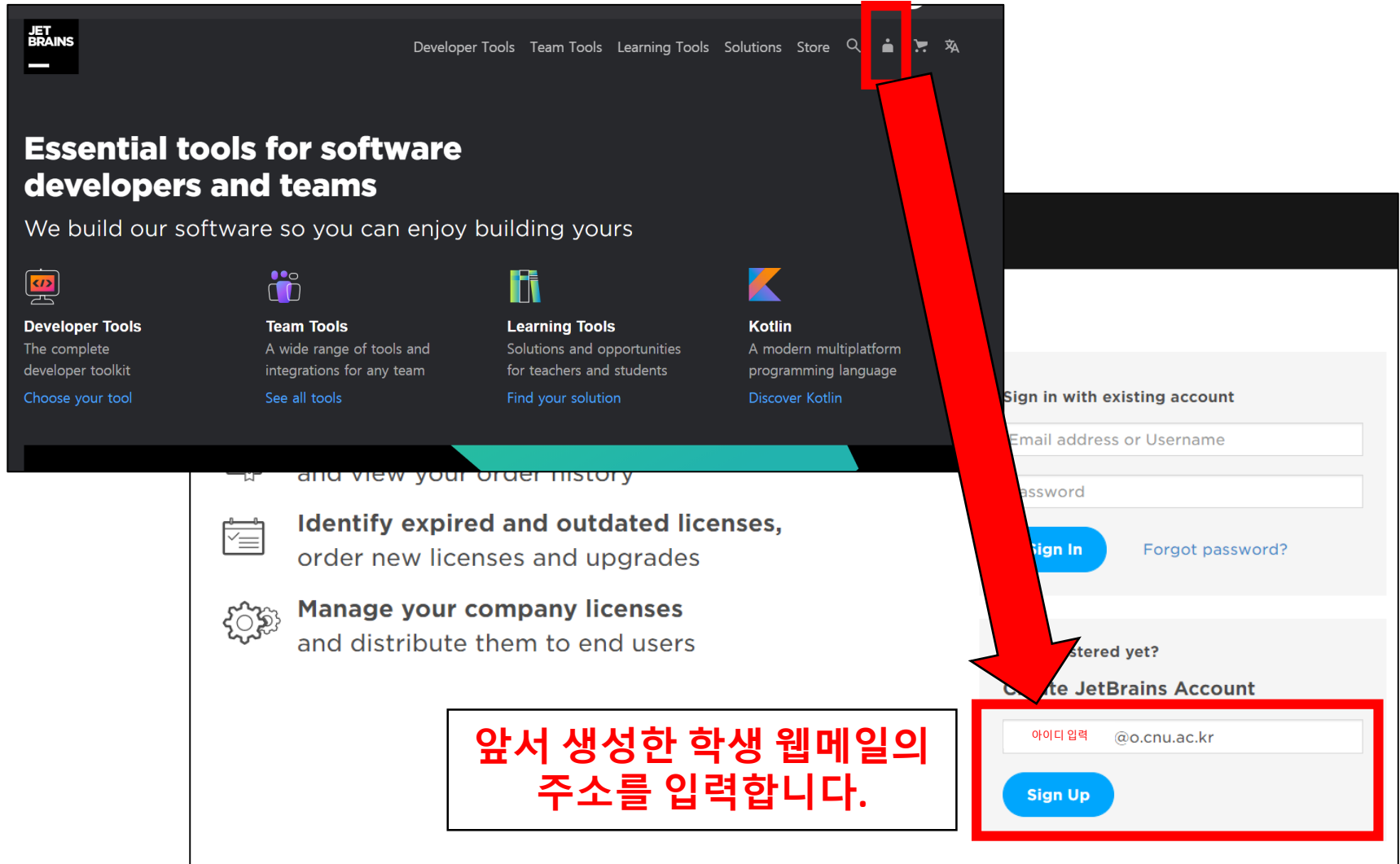
PyCharm 설치

2. JetBrains(<https://www.jetbrains.com/>) 홈페이지에 접속



PyCharm 설치

3. JetBrains 회원 가입



The screenshot shows the JetBrains website's homepage. The top navigation bar includes links for Developer Tools, Team Tools, Learning Tools, Solutions, and Store. A user icon in the top right corner is highlighted with a red box. A large red arrow points from this icon to the 'Create JetBrains Account' section on the right side of the page. In this section, the 'Email address or Username' input field is pre-filled with the text '아이디 입력 @o.cnu.ac.kr'. The 'Sign Up' button is highlighted with a red box. Below the input field, there is a 'Sign In' button and a link for 'Forgot password?'. The 'Create JetBrains Account' section also includes a 'Registered yet?' link. On the left side of the page, there are sections for 'Developer Tools', 'Team Tools', 'Learning Tools', and 'Kotlin'. A red box highlights the '아이디 입력' text in the email input field.

Essential tools for software developers and teams

We build our software so you can enjoy building yours

- Developer Tools**
The complete developer toolkit
[Choose your tool](#)
- Team Tools**
A wide range of tools and integrations for any team
[See all tools](#)
- Learning Tools**
Solutions and opportunities for teachers and students
[Find your solution](#)
- Kotlin**
A modern multiplatform programming language
[Discover Kotlin](#)

and view your order history

- Identify expired and outdated licenses,**
order new licenses and upgrades
- Manage your company licenses**
and distribute them to end users

Sign in with existing account

Email address or Username

Password

[Sign In](#) [Forgot password?](#)

Create JetBrains Account

아이디 입력 @o.cnu.ac.kr

[Sign Up](#)

앞서 생성한 학생 웹메일의 주소를 입력합니다.

PyCharm 설치

4. 학교 웹메일 계정으로 접속하여 발송된 메일 확인

The screenshot shows the Outlook webmail interface for Chungnam National University (CNU). The left sidebar contains navigation options like '즐거찾기' (Favorites), '받은 편지함' (Inbox), '보낸 편지함' (Sent), '임시 보관함' (Drafts), '즐거찾기에 추가' (Add to Favorites), '폴더' (Folders), and '받은 편지함' (Inbox). The main area displays an email from 'JetBrains Account' with the subject 'Complete your account registration'. The email content includes a greeting, a thank you message, and a link to 'Confirm your account'. The link is highlighted with a red box. Below the email content, there are buttons for 'Did you do this?', 'Did you request this?', and 'What is this?'. At the bottom, there are links for '회신' (Reply) and '전달' (Forward).

Complete your account registration

보낸 사람이 수신 허용 - 보낸 사람 목록에 없으므로 이 메시지 일부 내 차단된 콘텐츠 표시

메시지를 한국어(으)로 번역 | 영어에서 번역하지 않음

JetBrains Account <no_reply@jetbrains.com>
일 2021-03-07 오후 6:29
받는 사람: 허세훈

Hello!

Thank you for creating your JetBrains Account.

To complete your registration, please click the link below:

[Confirm your account](https://www.jetbrains.com/confirm)

Yours truly,
The JetBrains Team
<https://www.jetbrains.com>
The Drive to Develop

Did you do this? Did you request this? What is this?

위의 제안이 도움이 되었나요? 예 아니요

회신 | 전달

**“Confirm your account”
를 클릭하여 회원가입을
합니다.**

PyCharm 설치

5. 학생 인증 받기 : <https://www.jetbrains.com/community/education/#students> 에 접속

Free License Programs

[Academic Licensing](#) [Open Source](#) [User Groups](#) [Events Partnership](#) [Developer Recognition](#)

Free Educational Licenses

Learn or teach coding with best-in-class development tools from JetBrains!

[For students and teachers](#) [For schools and universities](#) [For training courses and bootcamp](#)

Individual licenses for students and teachers

Get free access to all JetBrains IDEs for personal use at school or at home.

Who can get free individual licenses for education

Students and faculty from accredited educational institutions (high schools, colleges, and universities) are welcome to apply.

Students need to be enrolled in an accredited educational program that takes one or more years of full-time study to complete.

Not sure about the license terms? [Check out the FAQ](#) or read the full terms [here](#).

Apply now

PyCharm 설치

6. 학생 License 인증 확인

JetBrains에 로그인하면 다음과 같이
학생 라이선스를 받았음을 확인할 수 있습니다.

The screenshot shows the JetBrains account page for a user named 'Sehun Hu'. The page is titled '1 License' and features a 'Buy new license' button. The main section displays the 'JetBrains Product Pack for Students' with a 'Download' link. It lists the license details: 'Licensed to: Sehun Hu', 'License restriction: For educational use only', and 'Valid through: March 06, 2022'. A list of included products is shown, including AppCode, CLion, DataGrip, dotCover, dotMemory, dotTrace, GoLand, IntelliJ IDEA Ultimate, PhpStorm, PyCharm, ReSharper, ReSharper C++, Rider, RubyMine, and WebStorm. A blue button prompts the user to 'Please review License Agreement' and accept it to start using the licenses. At the bottom, a note suggests linking past purchases to the account by providing a license key or domain.

1 License [Buy new license](#)

JetBrains Product Pack for Students [Download](#) License ID: 라이선스 번호

Licensed to: **Sehun Hu**
License restriction: For educational use only
Valid through: March 06, 2022
Following products included:

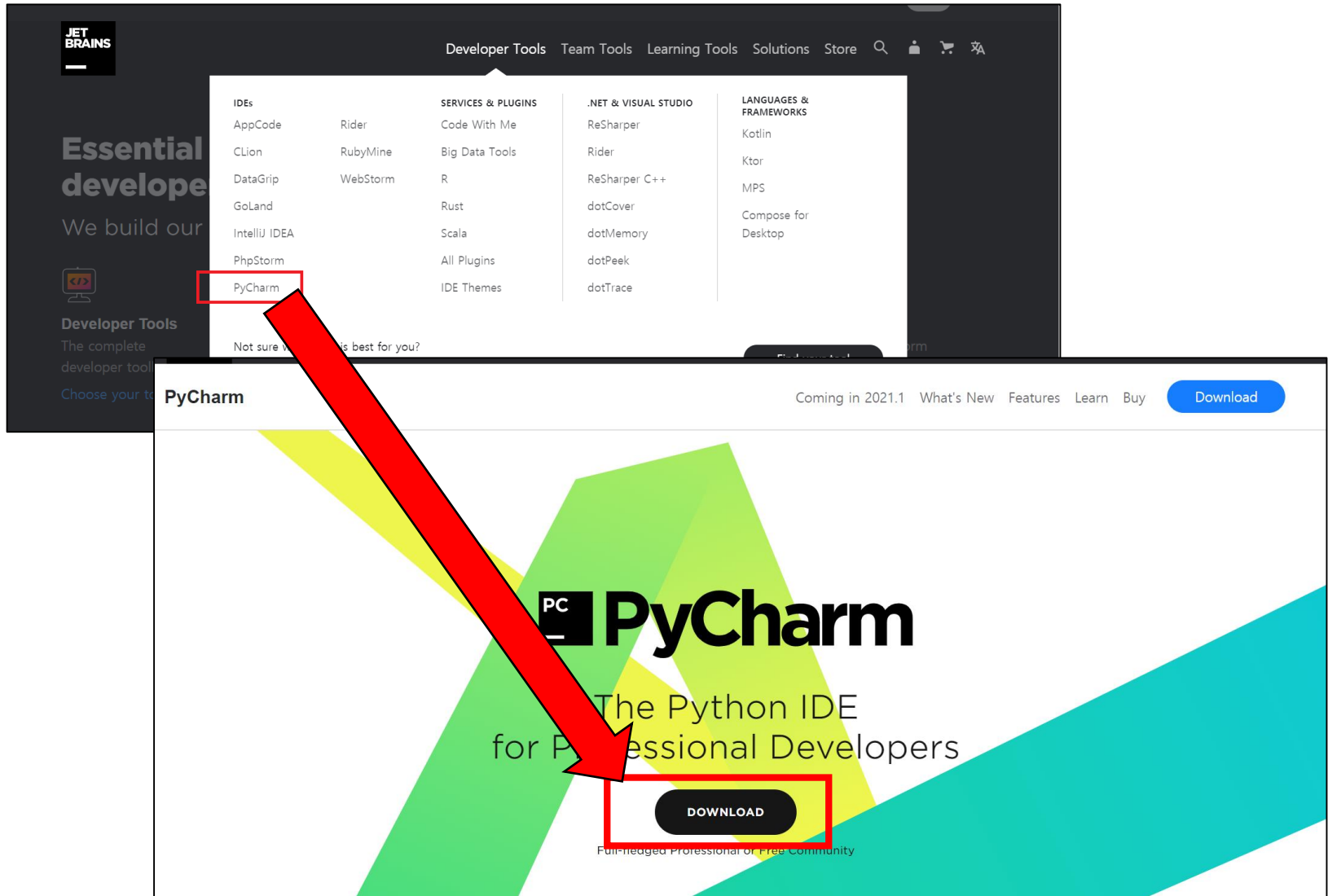
- AppCode
- CLion
- DataGrip
- dotCover
- dotMemory
- dotTrace
- GoLand
- IntelliJ IDEA Ultimate
- PhpStorm
- PyCharm
- ReSharper
- ReSharper C++
- Rider
- RubyMine
- WebStorm

[Please review License Agreement](#) and accept it to start using your licenses.

Can't find your license here? Link your past purchases to your JetBrains Account by [providing a license key or domain](#).

PyCharm 설치

7. Developer Tools → PyCharm 클릭 → DOWNLOAD 버튼 클릭




PyCharm 설치

8. 자신의 운영체제 선택 → Professional 버전 Download

PyCharm

Coming in 2021.1 What's New Features Learn Buy



Version: 2020.3.3
Build: 203.7148.72
27 January 2021

[System requirements](#)
[Installation Instructions](#)
[Other versions](#)

Download PyCharm

[Windows](#) macOS Linux

Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

[Download](#)


Free trial

Community

For pure Python development

[Download](#)

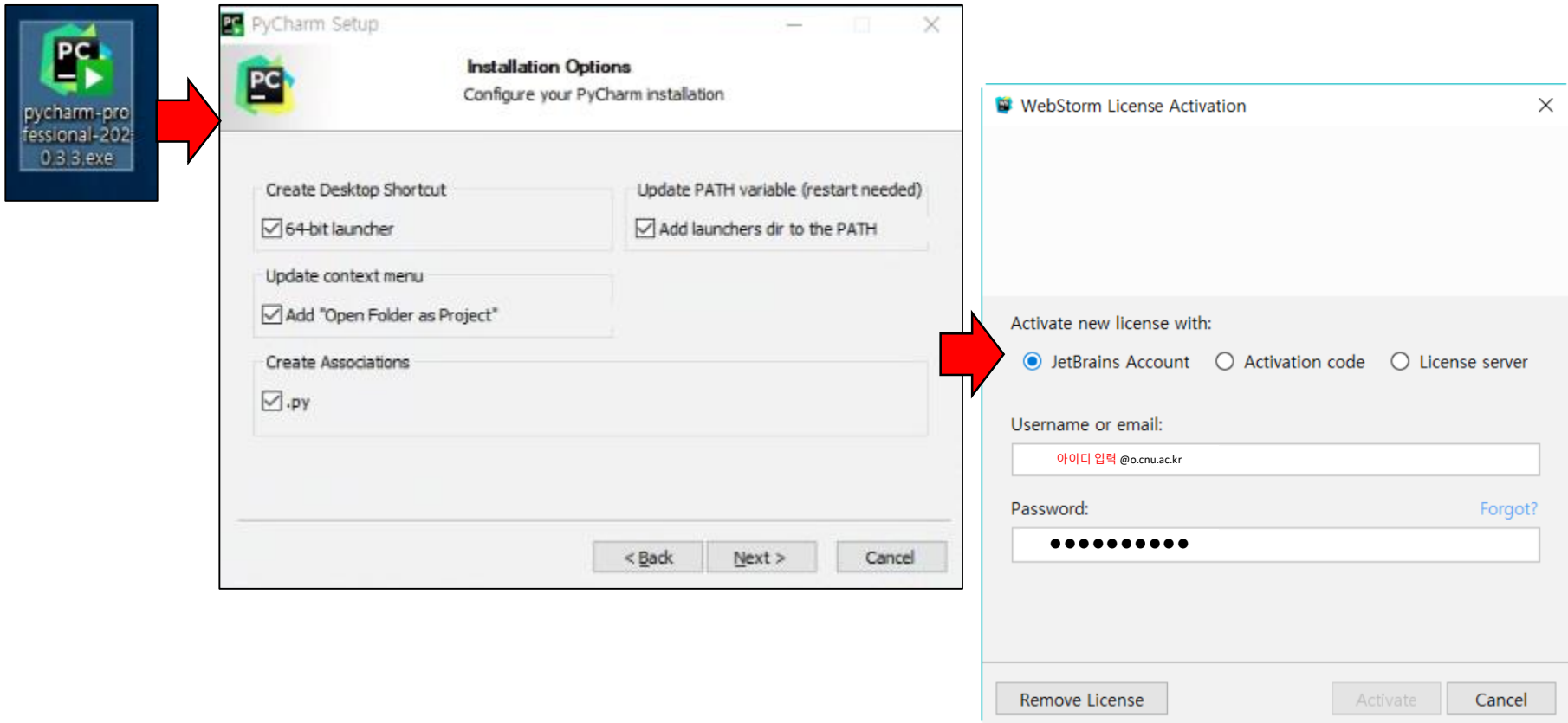
Free, open-source



Get the Toolbox App to download PyCharm and its future updates with ease

PyCharm 설치

9. 설치 및 라이선스 입력





수많은 데이터 과학
패키지 제공

Numpy, Pandas, Scikit-learn과 같은 데이터 과학을 위한 패키지들을 제공

패키지 및 종속성 관리

Conda 가상환경을 통해, 프로젝트들 간의 패키지 및 종속성을 구분해서 관리 가능

Anaconda | 설치

1. 아나콘다 설치 페이지(<https://www.anaconda.com/products/individual#download-section>)로 이동 → 자신에 맞는 Installer 파일 다운로드

Anaconda Installers

Windows

Python 3.8

64-Bit Graphical Installer (457 MB)

32-Bit Graphical Installer (403 MB)

MacOS

Python 3.8

64-Bit Graphical Installer (435 MB)

64-Bit Command Line Installer (428 MB)

Linux

Python 3.8

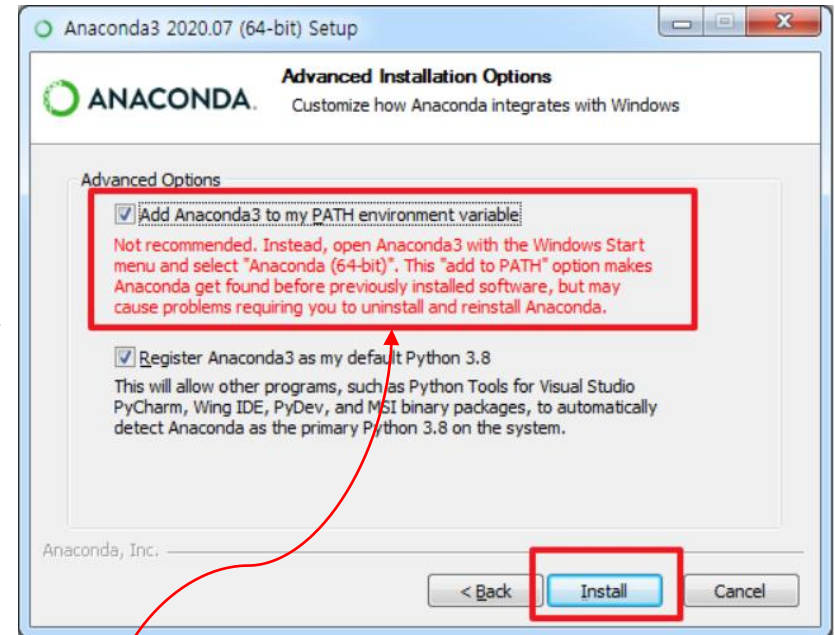
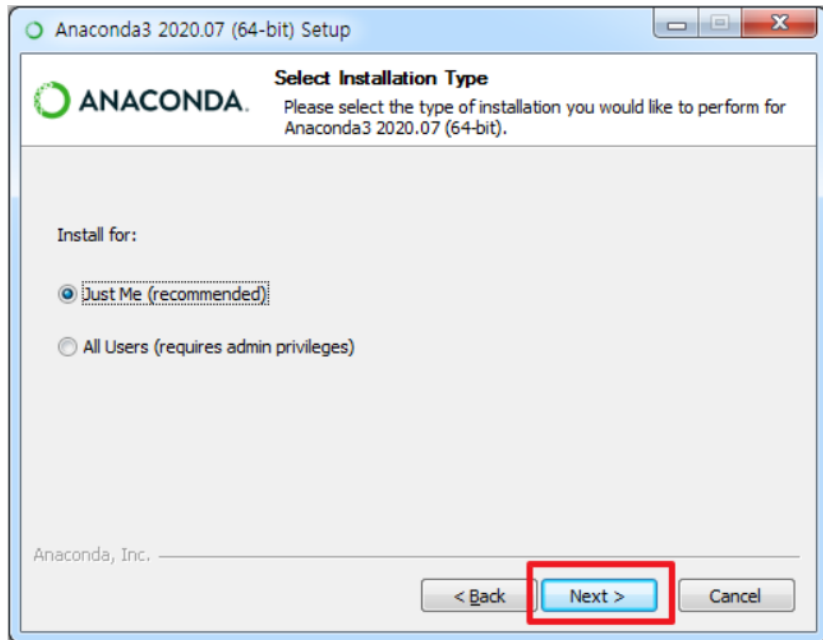
64-Bit (x86) Installer (529 MB)

64-Bit (Power8 and Power9) Installer (279 MB)

Windows 64비트의 경우

Anaconda | 설치

2. Anaconda 설치 프로그램 실행



**반드시 체크해주시기
바랍니다.**

“ 독립적인 작업환경에서 작업을 할 수 있게 도와줌 ”

프로젝트간의
패키지 버전 문제

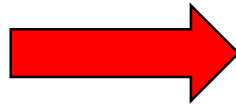
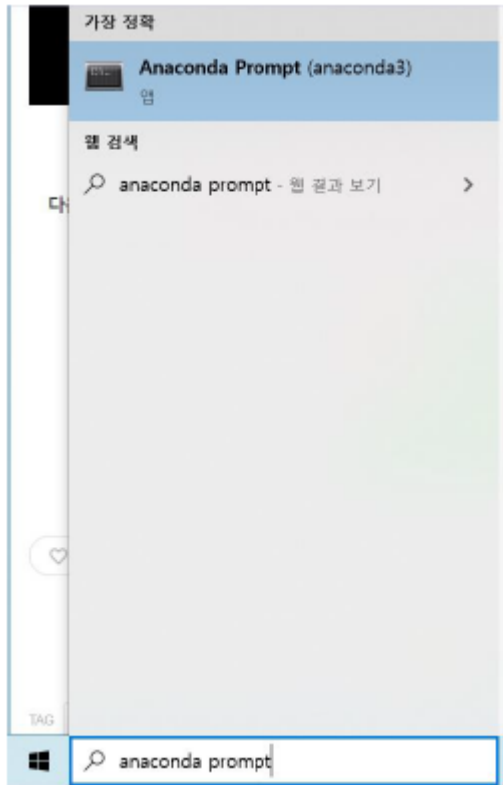
프로젝트마다 패키지의 버전들이 달라서
충돌이 발생하는 경우가 많다.

가상 환경을 통해
독자적으로 패키지 관리

Anaconda 가상 환경을 사용하면,
다른 프로젝트와 겹치지 않도록 할 수 있다.

Anaconda | 가상환경 생성 방법

1. Anaconda Prompt 실행 : Windows키를 누른 후, “anaconda prompt”라고 입력 후 실행



Anaconda | 가상환경 생성 방법

2. 가상환경 생성 명령어 입력

```
conda create -n {가상환경이름} python={Python 버전}
```

Anaconda Prompt (anaconda3)

```
(base) C:\Users\#tpgns>conda create -n ai python=3.8
```

Anaconda Prompt (anaconda3) - conda create -n ai python=3.8

The following packages will be downloaded:

package	build	
certifi-2020.12.5	py38haa95532_0	141 KB
pip-21.0.1	py38haa95532_0	1.8 MB
python-3.8.8	hdbf39b2_4	15.9 MB
setuptools-52.0.0	py38haa95532_0	726 KB
Total:		18.6 MB

The following NEW packages will be INSTALLED:

ca-certificates	pkgs/main/win-64::ca-certificates-2021.1.19-haa95532_0
certifi	pkgs/main/win-64::certifi-2020.12.5-py38haa95532_0
openssl	pkgs/main/win-64::openssl-1.1.1j-h2bbff1b_0
pip	pkgs/main/win-64::pip-21.0.1-py38haa95532_0
python	pkgs/main/win-64::python-3.8.8-hdbf39b2_4
setuptools	pkgs/main/win-64::setuptools-52.0.0-py38haa95532_0
sqlite	pkgs/main/win-64::sqlite-3.33.0-h2a8f88b_0
vc	pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime	pkgs/main/win-64::vs2015_runtime-14.27.29016-h555637f_2
wheel	pkgs/main/noarch::wheel-0.36.2-pyhd3eb1b0_0
wincertstore	pkgs/main/win-64::wincertstore-0.2-py38_0
zlib	pkgs/main/win-64::zlib-1.2.11-h62dd497_4

Proceed ([y]/n)

y

“y”를 입력 후,
Enter키를 누르면
가상환경이 생성됩니다.

Anaconda | 가상환경 활성화/비활성화

1. 가상환경 활성화

```
conda activate {가상환경 이름}
```

```
Anaconda Prompt (anaconda3)
(base) C:\Users\User>conda activate ai
(ai) C:\Users\User>
```

“ai”라는 가상환경으로 변경됨

2. 가상환경 비활성화

```
conda deactivate
```

```
Anaconda Prompt (anaconda3) - conda deactivate
(ai) C:\Users\User>conda deactivate
(base) C:\Users\User>
```

기본 가상환경으로 돌아옴

Anaconda | 가상환경 관련 명령어들

	명령어 형태	예시
가상환경 생성	conda create -n {가상환경이름} python={python 버전}	conda create -n AI python=3.8
가상환경 목록 확인	conda env list	conda env list
가상환경 활성화	conda activate {가상환경 이름}	conda activate AI
가상환경 비활성화	conda deactivate {가상환경이름}	conda deactivate AI
가상환경 삭제	conda env remove -n {가상환경이름}	conda env remove -n AI
가상환경 내에서 패키지 설치	conda install {패키지 이름}	conda install numpy

Jupyter Notebook | 소개



코드의 실행 결과를
즉각적으로 확인 가능

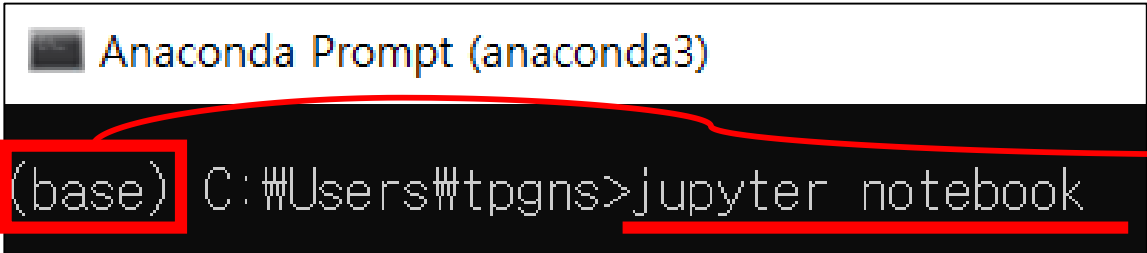
코드를 즉각적으로 실행함으로써
데이터를 분석하거나 시각화할 수 있음

커뮤니케이션 또는
문서화 가능

Jupyter Notebook을 통해 생성되는 .ipynb 파일을
공유하거나 문서화하여 의사소통 가능

Jupyter Notebook | 실행

1. Anaconda Prompt에서 “jupyter notebook”이라는 명령어를 입력



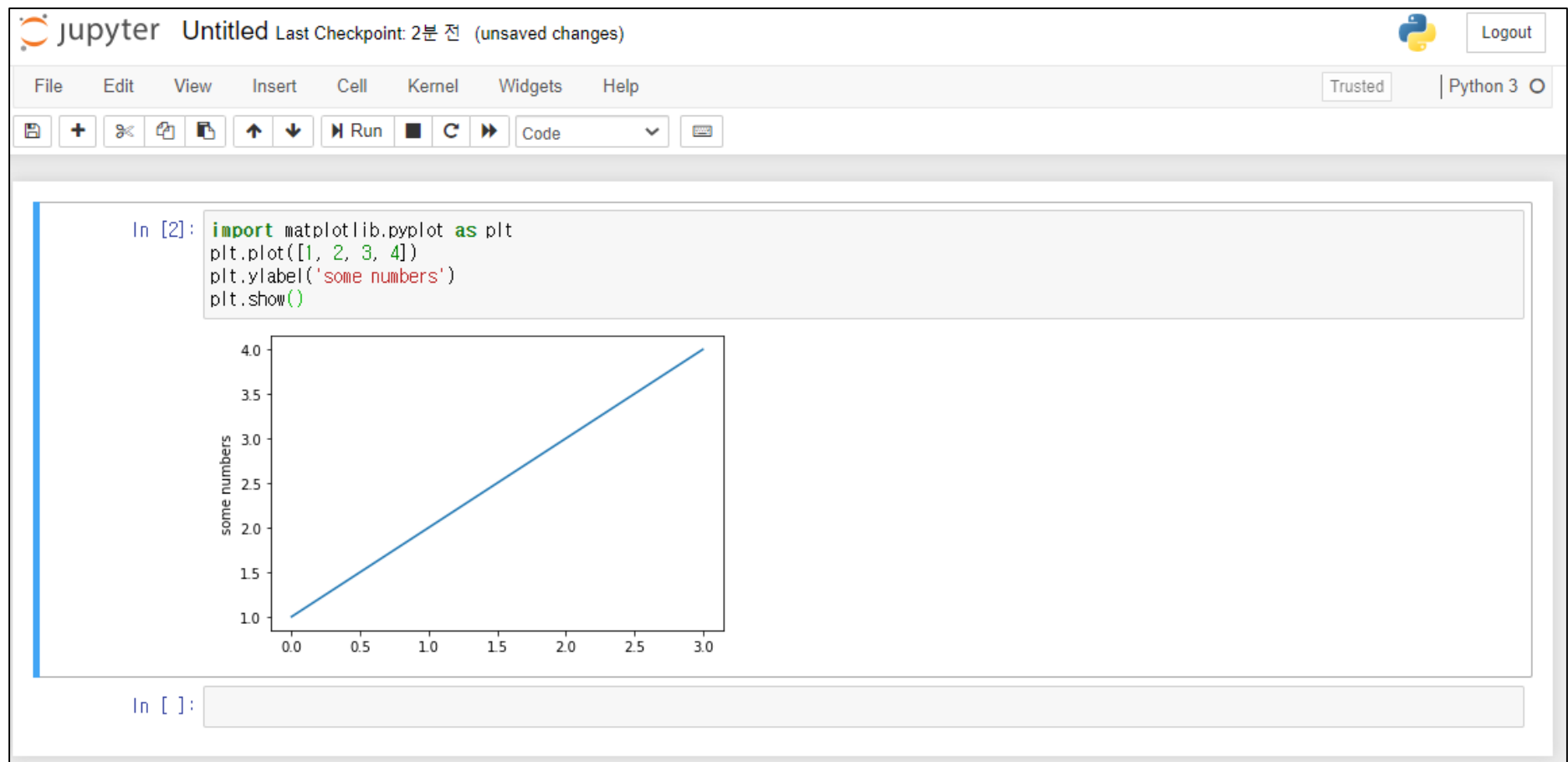
반드시 기본 가상환경
에서 실행해주세요.

2. 쥬피터 노트북 파일 생성



Jupyter Notebook | 예시

나만의 코드 실행하기



Numpy | 소개



다차원 배열 자료형 및
연산 제공

Python에는 “리스트”라는 자료형은 있지만
“배열”은 없다. → Numpy는 배열 자료형을 제공

다양한 수학적 연산
제공

수학적인 연산을 Function 형태로 제공

Numpy | 설치 및 사용

1. Anaconda Prompt에서 “conda install numpy”이라는 명령어를 입력

Anaconda Prompt (anaconda3) - conda deactivate

```
(ai) C:\Users\User>conda install numpy
```

Anaconda Prompt (anaconda3) - conda deactivate - conda install numpy

The following packages will be downloaded:

package	build	size
intel-openmp-2020.2	254	1.6 MB
mk1-2020.2	256	109.3 MB
mk1-service-2.3.0	py38h196d8e1_0	47 KB
mk1_fft-1.3.0	py38h46781fe_0	134 KB
numpy-1.19.2	py38hadc3359_0	22 KB
numpy-base-1.19.2	py38ha3acd2a_0	3.8 MB
six-1.15.0	py38haa95532_0	27 KB
Total:		114.9 MB

The following NEW packages will be INSTALLED:

blas	pkgs/main/win-64::blas-1.0-mkl
intel-openmp	pkgs/main/win-64::intel-openmp-2020.2-254
mk1	pkgs/main/win-64::mk1-2020.2-256
mk1-service	pkgs/main/win-64::mk1-service-2.3.0-py38h
mk1_fft	pkgs/main/win-64::mk1_fft-1.3.0-py38h4678
mk1_random	pkgs/main/win-64::mk1_random-1.1-py38h4
numpy	pkgs/main/win-64::numpy-1.19.2-py38hadc33
numpy-base	pkgs/main/win-64::numpy-base-1.19.2-py38h
six	pkgs/main/win-64::six-1.15.0-py38haa95532_

Proceed ([y]/n)? y

이전에 만들었던 가상환경을 활성화한 후에 입력해주세요.

“y” 입력

2. import를 통해 numpy를 사용

```
import numpy as np # numpy 라이브러리를 np라는 이름으로 import합니다.  
  
arr = np.array([1, 2, 3, 4])  
print(arr)
```

Numpy | 배열 생성 방법

(1) Python 리스트 → Numpy 배열

```
import numpy as np

list_python = [1, 2, 3, 4] # 파이썬 리스트를 만듭니다.
numpy_arr = np.array(list_python) # 파이썬 리스트를 numpy 배열로 만듭니다.
print(numpy_arr.shape) # (4, )
```

(2) Numpy에서 제공하는 함수를 통해 Numpy 배열 생성하기

```
import numpy as np

zero_arr = np.zeros((2, 2)) # 전부 0으로 채워진 (2, 2)짜리 배열 생성
one_arr = np.ones((2, 3)) # 전부 1로 채워진 (2, 3)짜리 배열 생성
five_arr = np.full((2, 3), 5) # 전부 5로 채워진 (2, 3)짜리 배열 생성
identity_arr = np.eye(3) # (3, 3)짜리 단위 행렬 생성
```

Numpy | 배열 슬라이싱

“지정한 범위가 아닌 원소들을 잘라냄”

```
import numpy as np

p_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
arr = np.array(p_list) # Python 리스트로부터 (3, 3) 짜리 배열 생성

ex1 = arr[0:2, 0:2] # row와 column이 0~1까지인 원소들만 선택
print(ex1)

ex2 = arr[1: , 1:] # row와 column이 1~끝까지인 원소들만 선택
print(ex2)

ex3 = arr[-1, 0:] # row가 마지막이고 column이 0~끝까지인 원소들만 선택
print(ex3)
```

Numpy | 배열 인덱싱

“배열의 인덱스를 통해 원하는 원소들만 선택”

(1) 자신이 원하는 원소들을 Index로 접근해서 가져오기

```
import numpy as np

target_arr = np.array([11, 22, 33, 44, 55, 66, 77, 88, 99])
idx = np.array([0, 2, 4, 6, 8]) # 선택하고 싶은 Index들을 np.array 형태에 저장합니다.
print(target_arr[idx]) # [11, 33, 55, 77, 99]
```

[주의] Index를 위한 배열은 Numpy Array 형태이어야 함

(2) Boolean 값으로 인덱싱하기

```
import numpy as np

target_arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
bool_idx = (target_arr % 2 == 0) # 2로 나누어 떨어지는 지를 boolean 값으로 저장합니다.
print(bool_idx) # [True, False, True, False, True, False, True, False, True, False]
print(target_arr[bool_idx]) # [0, 2, 4, 6, 8]
```

[한줄로도 작성 가능] target_arr[target_arr % 2 == 0]

Numpy | 배열의 크기 및 자료형

(1) 배열 크기 확인

```
import numpy as np

a = np.array([1, 2, 3])
print(a.shape) # (3, )

b = np.array([[1, 2, 3], [4, 5, 6]])
print(b.shape) # (2, 3)
```

shape가 (3,)인 것과 (3, 1)은 완전히
다른 의미입니다.
(3, 1)은 np.array([[1], [2], [3]])을 의미

(2) 배열 자료형 확인

```
import numpy as np

a = np.array([1, 2, 3])
print(a.dtype) # dtype('int64')

b = np.array([1, 2, 3.])
print(b.dtype) # dtype('float64')
```

numpy 자료형

arr1이나 arr2는 int64라는 자료형을 갖는 것에 반해 arr3는 float64라는 자료형을 갖는다.

이는 arr3내부 데이터를 살펴보면 3.5라는 실수형 데이터를 갖기 때문임을 알 수 있다.

numpy에서 사용되는 자료형은 아래와 같다. (자료형 뒤에 붙는 숫자는 몇 비트 크기인지를 의미한다.)

- 부호가 있는 정수 int(8, 16, 32, 64)
- 부호가 없는 정수 uint(8, 16, 32, 64)
- 실수 float(16, 32, 64, 128)
- 복소수 complex(64, 128, 256)
- 불리언 bool
- 문자열 string_
- 파이썬 오브젝트 object
- 유니코드 unicode_

Numpy | 배열끼리의 연산

(1) 기본적으로는
크기가 서로 동일할 때 연산 가능

```
import numpy as np

a = np.array([1, 2, 3])
b = np.array([4, 5, 6])

c = a + b # 각 원소끼리 더하기
print(c)

c = a - b # 각 원소끼리 빼기
print(c)

c = a * b # 각 원소끼리 곱하기
print(c)

c = a / b # 각 원소끼리 나누기
print(c)
```

행렬 곱
X

(2) 브로드캐스트라는 기능을 통해,
연산 가능

```
import numpy as np

a = np.array([[1, 2, 3], [4, 5, 6]])
print(a.shape)

b = np.array([10, 11, 12])
print(b.shape)

print(a + b) # array([[11, 13, 15], [14, 16, 18]])
print(a * b) # array([[10, 22, 36], [40, 55, 72]])

print(a * 10) # array([[10, 20, 30], [40, 50, 60]])
print(a ** 2) # array([[ 1,  4,  9], [16, 25, 36]])
```

* 브로드캐스트 : 서로 크기가 다른
배열끼리 연산이 가능하도록 하는 기능

Numpy | 수학 연산 함수들

(1) 하나의 배열/수에 적용되는 함수들

함수	설명	예시
abs()	절대값 계산	np.abs(arr1)
sqrt()	제곱근 계산	np.sqrt(arr1)
square()	제곱 계산	np.square(arr1)
exp()	e의 지수로 삼은 값 계산	np.exp(arr1)
log()	로그 취하기	np.log(arr1)
sign()	부호 계산	np.sign(arr1)
ceil()	소수 첫 번째 자리에서 올림	np.ceil(arr1)
isnan()	NaN 값인 경우, True, 아닌 경우 False 반환	np.isnan(arr1)
sin()	사인 함수	np.sin(arr1)
cos()	코사인 함수	np.cos(arr1)

Numpy | 수학 연산 함수들

(2) 두 개의 배열에 적용되는 함수들

함수	설명	예시
add()	동일한 위치의 성분끼리 더하기	np.add(arr1, arr2)
subtract()	동일한 위치의 성분끼리 빼기	np.subtract(arr1, arr2)
multiply()	동일한 위치의 성분끼리 곱하기	np.multiply(arr1, arr2)
divide()	동일한 위치의 성분끼리 나누기	np.divide(arr1, arr2)
maximum()	동일한 위치의 성분끼리 비교하여 최대값 계산하기	np.maximum(arr1, arr2)
minimum()	동일한 위치의 성분끼리 비교하여 최소값 계산하기	np.minimum(arr1, arr2)
dot()	행렬 곱	np.dot(arr1, arr2)

Numpy | 수학 연산 함수들

(3) 통계 관련 함수들

함수	설명	예시
sum()	행렬 내의 성분 합 계산	<ul style="list-style-type: none">np.sum(arr1) → 전체 성분의 합np.sum(arr1, axis=0) → 행 간의 합np.sum(arr1, axis=1) → 열 간의 합
mean()	행렬 내의 성분 평균 계산	<ul style="list-style-type: none">np.mean(arr1) → 전체 성분의 평균np.mean(arr1, axis=0) → 행 간의 평균np.mean(arr1, axis=1) → 열 간의 평균
std()	행렬 내의 표준편차 계산	np.std(arr1)
var()	행렬 내의 분산 계산	np.var(arr1)
argmax()	행렬 내의 최대값이 위치한 인덱스 반환	<ul style="list-style-type: none">np.argmax(arr1)np.argmax(arr1, axis=0)
argmin()	행렬 내의 최소값이 위치한 인덱스 반환	<ul style="list-style-type: none">np.argmin(arr1)np.argmin(arr1, axis=0)

Numpy | 정렬 관련 함수들

- `sort()` : 정렬된 값을 반환
- `argsort()` : 정렬된 값의 인덱스를 반환

```
import numpy as np

a = np.array([2, 5, 3, 6, 8, 1])

b = np.sort(a) # 정제 성분에 대해서 오름차순으로 정렬
print(b)

c = np.sort(a)[::-1] # 전체 성분에 대해서 내림차순으로 정렬
print(c)

d = np.argsort(a) # 오름차순 순으로 인덱스를 반환
print(d)
```

Numpy | 난수 생성

- Numpy의 random 모듈 사용
- rand() 함수에 크기를 입력해준다.

```
import numpy as np

a = np.random.rand(5)
print(a)

b = np.random.rand(2, 3)
print(b)
```