Algorithm 으로 배우는 Python

강사: 오영제

Python 의 역사

- 1991 년 Guido Van Rossum (네덜란드) 이 개발
  - Open Source
  - Scripting Language – No compiler
  - Object Oriented 와 Functional Programming 모두 지원
- Web programming 에서 Machine Learning 까지 다양한 분야에 사용
- Python 이름은 snake 가 아니라 drama 제목에서 따옴
- Python Version
  - Version 2.7 – 마지막 2.0 버전 (2010) → 아직 사용
  - Version 3.8 – 현재

타언어와의 비교

	Python	C/Java	언어 선정 고려점
• 학습 속도	Python > Java > C		
• Program 수행 속도	C > Java > Python		과거– CPU 속도 및 가격 현재 – 가독성 / 유지보수 용이성
• 개발 생산성	Python > Java / C		
• 유지 보수성	Python > Java / C		
• OOP	지원	지원	
• Functional P	지원	부분적 지원	
• Compile	없음	Compile 후 실행	Python 은 Script 언어
• 적용 분야	Web 에서 Machine Learning 까지 전 분야	Web 개발의 주류 Mobile Application 등	과학기술, Data Science, Machine Learning 등
• 타언어와의 연결성	Python 의 별명은 glue-language LPACK, Fortran, C++ 등과 interface		대부분 task 에서 충분히 빠름

Coding Style 비교

- **C**

```
#include <studio.h>
int main(int argc, char ** argv) {
        printf(“Hello, World!\n”);
}
```

- **Java**

```
public class Hello{
        public status void main(String argv[]){
                system.out.println(“Hello, World!”);
        }
}
```

- **Python**

```
print(“Hello, World!”)
```

- { } 로 block 구분
- ; 로 statement 종료 구분
- 함수의 반환 자료형 명시

없음

코드 길이 비교 (구구단 출력)

C++

```
#include "stdafx.h"
#include <iostream>
#include <iomanip>

int main()
{
    int mul = 1;
    int x, y;
    for (x = 2; x <= 9; x++) {
        for (y = 1; y <= 9; y++) {
            mul = x * y;
            cout << setw(1) << x << "*" << y
              << "*" << setw(2) << mul << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Python

```
for x in range(2,10):
    for y in range(1,10):
        mul = x * y
        print("{}*{}={} ".format(x, y, mul),
                end="")
    print()
```

```
2*1=2 2*2=4 2*3=6 2*4=8 2*5=10 2*6=12 2*7=14 2*8=16
2*9=18 3*1=3 3*2=6 3*3=9 3*4=12 3*5=15 3*6=18 3*7=21
3*8=24 3*9=27 4*1=4 4*2=8 4*3=12 4*4=16 4*5=20
4*6=24 4*7=28 4*8=32 4*9=36 5*1=5 5*2=10 5*3=15
5*4=20 5*5=25 5*6=30 5*7=35 5*8=40 5*9=45 6*1=6
6*2=12 6*3=18 6*4=24 6*5=30 6*6=36 6*7=42 6*8=48
6*9=54 7*1=7 7*2=14 7*3=21 7*4=28 7*5=35 7*6=42
7*7=49 7*8=56 7*9=63 8*1=8 8*2=16 8*3=24 8*4=32
8*5=40 8*6=48 8*7=56 8*8=64 8*9=72 9*1=9 9*2=18
9*3=27 9*4=36 9*5=45 9*6=54 9*7=63 9*8=72 9*9=81
```

코드 길이 비교 (카카오 사례)

- 2017 카카오 블라인드 공채 코딩 테스트 언어별 통계

  - 자바 43%, C++ 36%, 파이썬 11%, 자바스크립트 8%

  - 코드 라인 수: C++ > 자바 > 자바스크립트 > 파이썬

- 2019 카카오 블라인드 공채 코딩 테스트 언어별 통계

  - 파이썬 60%, 자바 26%, C++ 4%, 자바스크립트 12%

Python 언어의 특징

1. 직관적 자료구조 (8 가지 자료형)

- String, Decimal, Float, Boolean + List, Tuple, Set, Dictionary

  List – array 와 유사 → [‘cat’, ‘dog’, ‘lion’, ‘king’]

  Tuple – list 와 유사하나 값을 변경할 수 없음 → (cat’, ‘dog’, ‘lion’)

  Set (집합) → list 와 유사하나 중복을 허용하지 않음

  Dictionary – key/value pair → {‘cat’: 1, ‘dog’: 2, ‘lion’: 3}

2. 영어 문장과 유사한 명령어 형식

C++/Java	Python
String name = “Bob”; system.out.println(name); cout << name;	name = “Bob” print(name)

for (int i = 1; i < 21; i++){
  system.out.println(i)
}
	for i in range(1, 21):
	  print(i)

3. One Line Coding (Pythonic Way)

C++/JavaScript	Python

int temp = x;
x = y;
y = temp;
	x, y = y, x

let lst1 = [ ]
for (int i=0; i<4; i++) {
let lst2 = [ ]
for (int j=0; j<4; i++) {
lst2.push(i + j*5)
lst1.push(lst2)
}
}
console.log(lst1)
	[[i + j*5 for j in range(4)] for i in range(4)]
	
	➔ [[0, 5, 10, 15], [1, 6, 11, 16], [2, 7, 12, 17], [3, 8, 13, 18]]

4. Simple Syntax

들여쓰기로 block 구분 ({ }, semi-colon 없음)

```
if  a == b:
    c += 1
    print(“Same”, c)
elif a == d:
    c -= 1
    print(“Different”, c)
    if final:
        print(“final”)
    else:
        print(“not final”)
else:
     return a, b, c
```

5. Dynamically Typed

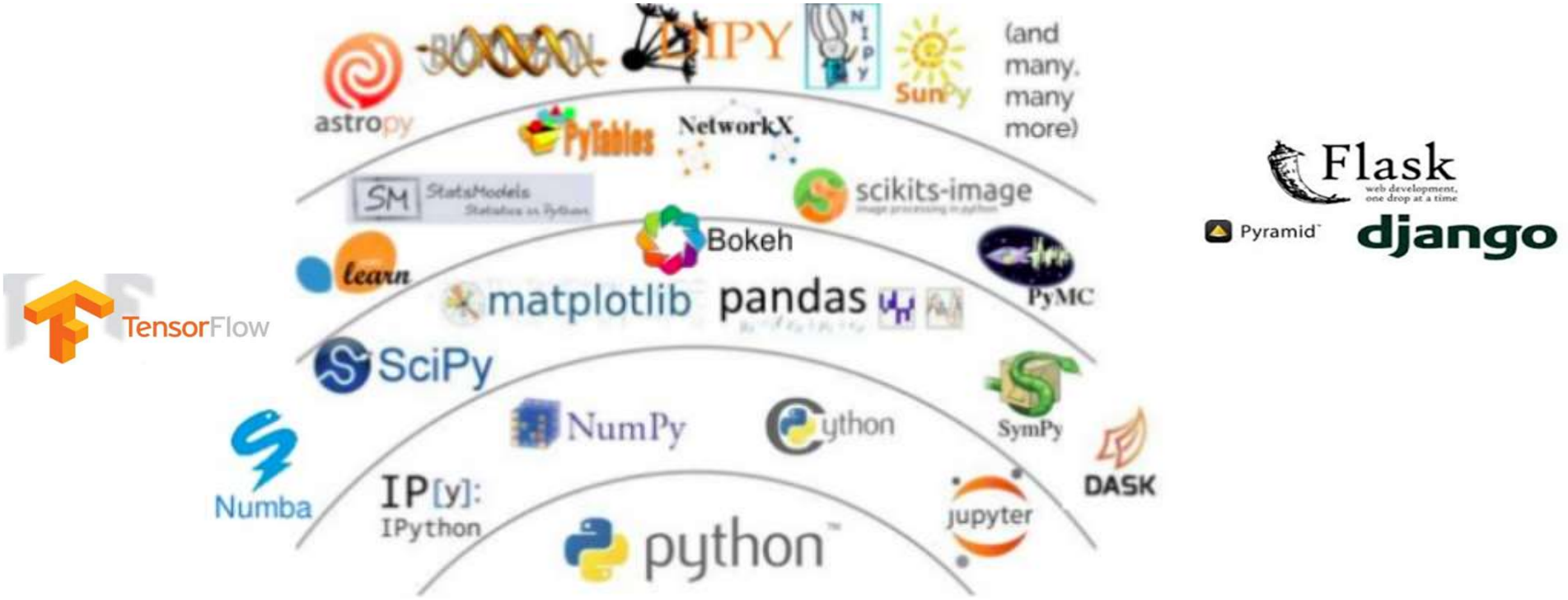
Java: int x = 1;
x = x / 2; ➔ x : 0
x = “abc” ➔ error

Python: x = 1
x = x / 2 ➔ x : 0.5 (integer 에서 float 으로 변환)
x = “abc” ➔ x : “abc” (float 에서 string 으로 변환)

6. Python 은 Interpreter 언어

- No compile required

7. 풍부한 Libraries (From Web to Data Science)



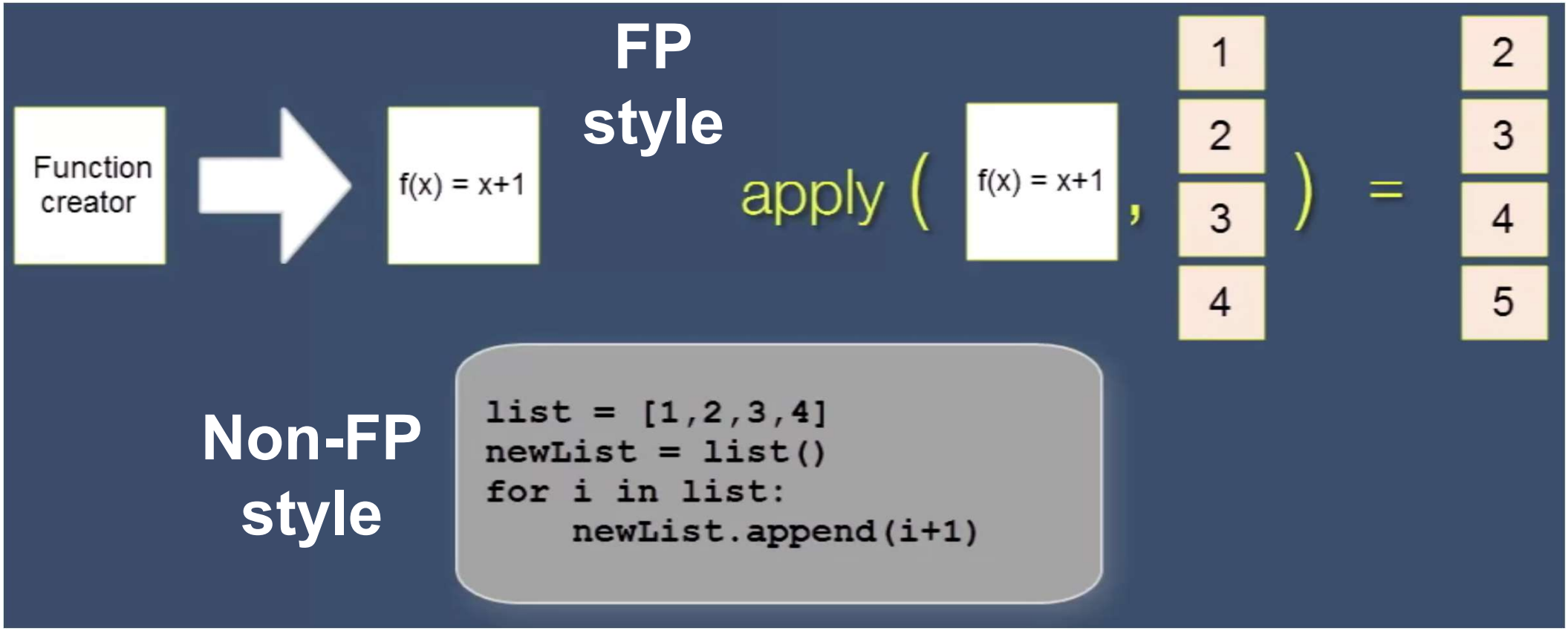
8. Object Oriented and Functional Programming 모두 지원

- OOP (Object Oriented Programming)
  - Class 를 이용한 객체 생성
  - 객체를 통한 method 호출

- Functional Programming (map, filter, reduce, lambda)
  - 함수 (Function) 정의
  - 입력값 처리 / 출력값 반환
  - 함수A → 함수B → 함수C

FP vs OOP

Data 병렬처리에 매우 유리



9. 다양한 내장함수

Built-in Functions				
abs()	divmod()	input()	open()	staticmethod()
all()	enumerate()	int()	ord()	str()
any()	eval()	isinstance()	pow()	sum()
basestring()	execfile()	issubclass()	print()	super()
bin()	file()	iter()	property()	tuple()
bool()	filter()	len()	range()	type()
bytearray()	float()	list()	raw_input()	unichr()
callable()	format()	locals()	reduce()	unicode()
chr()	frozenset()	long()	reload()	vars()
classmethod()	getattr()	map()	repr()	xrange()
cmp()	globals()	max()	reversed()	zip()
compile()	hasattr()	memoryview()	round()	__import__()
complex()	hash()	min()	set()	
delattr()	help()	next()	setattr()	
dict()	hex()	object()	slice()	
dir()	id()	oct()	sorted()	

실습환경 구축

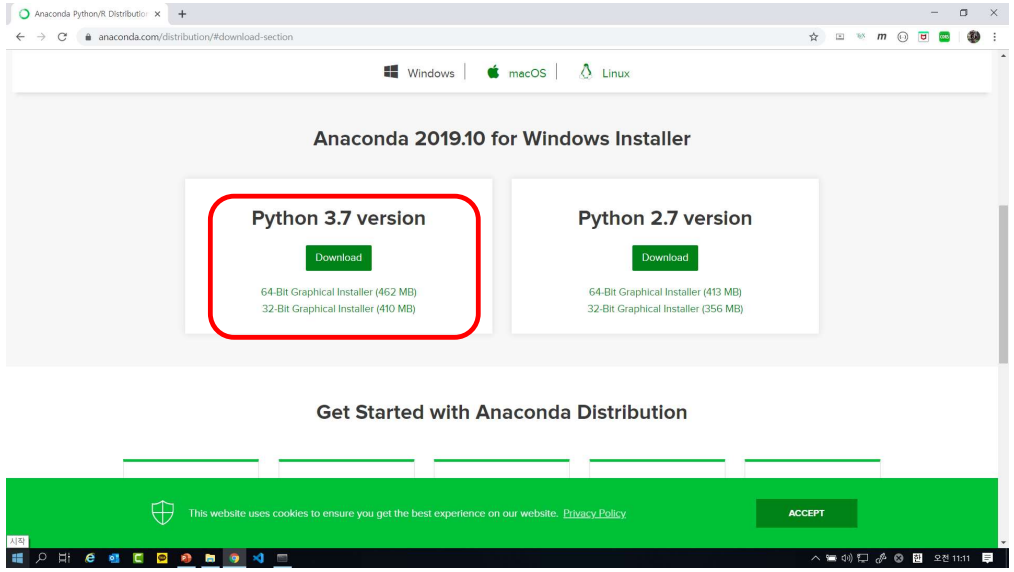
- Python 환경 (anaconda) 설치 방법

Python Language (Anaconda) 설치

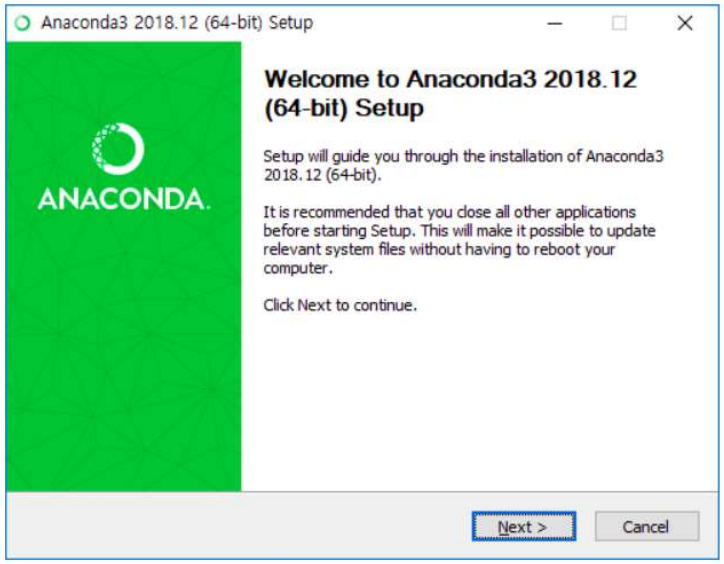
- 아나콘다는 여러가지 수학 및 과학 패키지(Package)들을 기본적으로 포함하고 있는 파이썬 배포판
- 파이썬은 오픈소스이므로 라이브러리들이 매우 빈번하게 업데이트 됨.
- 각각의 라이브러리들의 다양한 version 간 의존성을 배포 패키지에 반영
- 프로젝트 별 가상환경 설정하여 다양한 version 관리 가능
- 운영체제 (Windows, Linux, Mac) 에 통일된 GUI 제공

Anaconda 설치

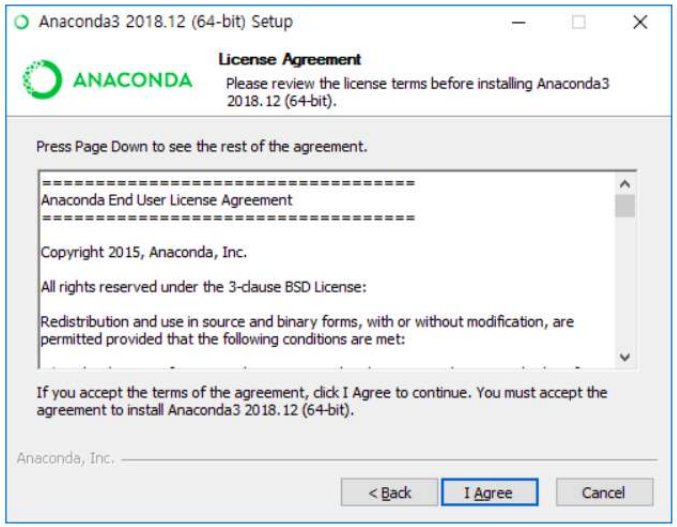
Download from
https://www.anaconda.com/distribution/#download-section

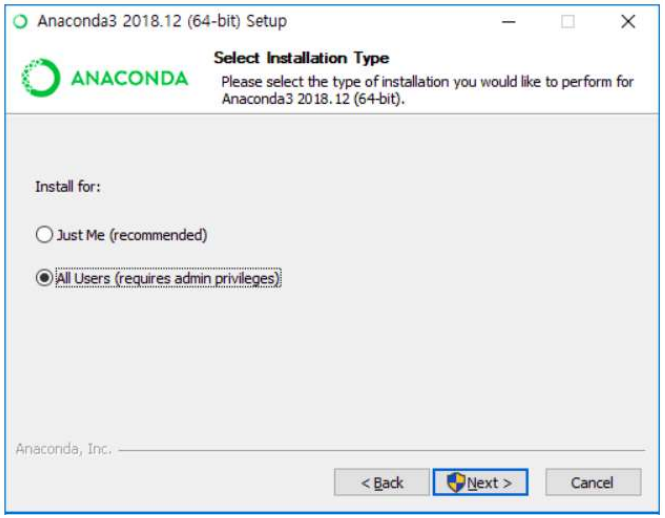


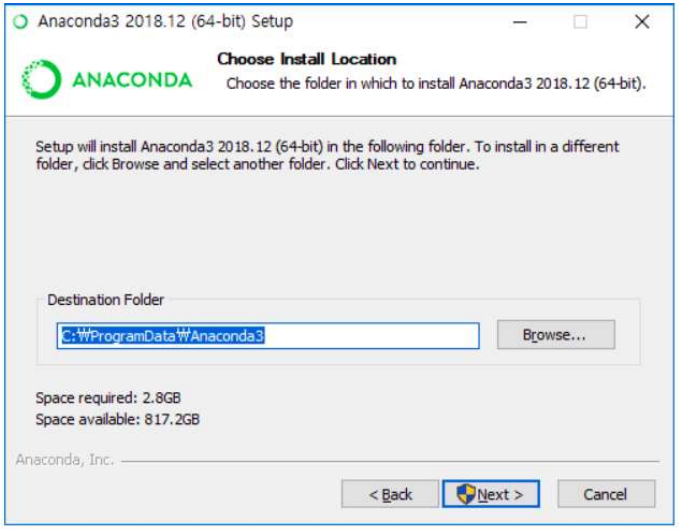
설치파일 실행

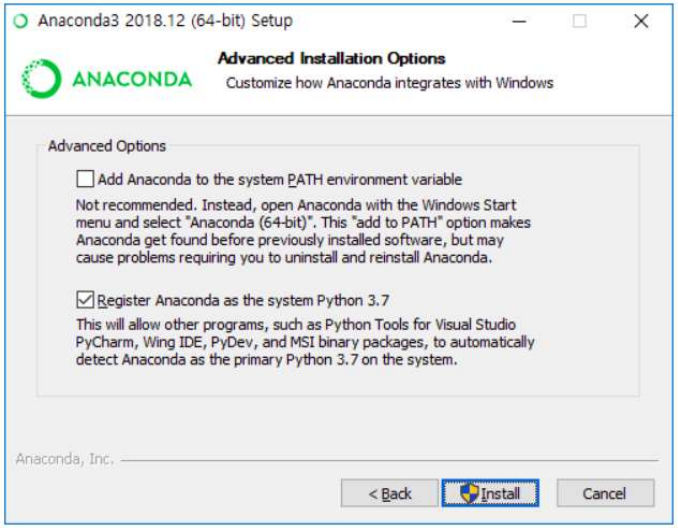


Next 를 눌러 설치 시작









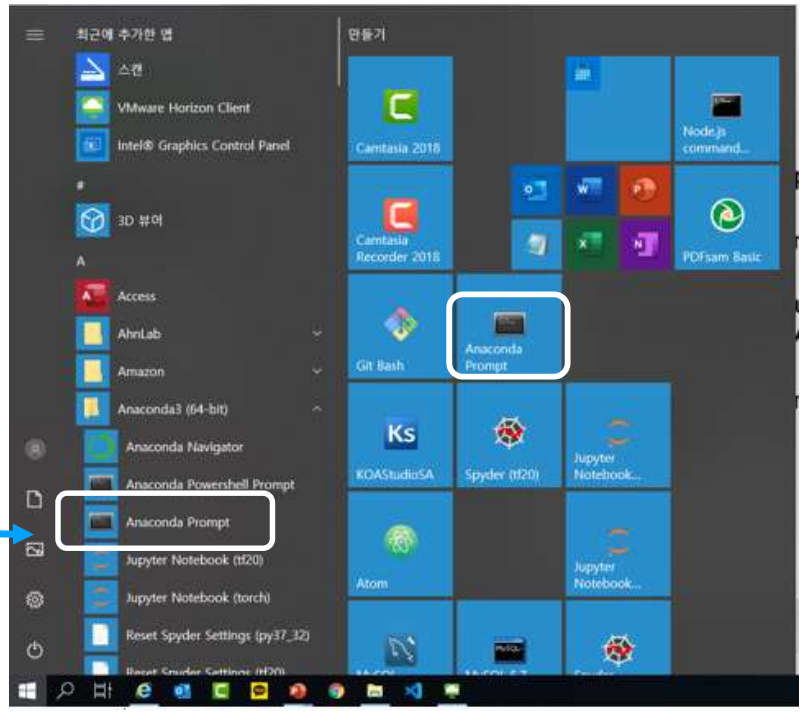
완료

Python 실행

1) Anaconda 명령 prompt 실행

2) Python 실행

(base) C:\Users\trimu>python



Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32

Type "help", "copyright", "credits" or "license" for more information.

>>>

Anaconda 가상환경 사용법

1) 가상환경 생성

conda create –n mytestenv python=3.7

```
(myenv) C:₩Users₩trimu>conda create -n mytestenv python=3.7
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: C:₩Users₩trimu₩Miniconda3₩envs₩mytestenv

  added / updated specs:
    - python=3.7


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    certifi-2020.4.5.1         |           py37_0         156 KB
    ------------------------------------------------------------
                                           Total:         156 KB

The following NEW packages will be INSTALLED:

  ca-certificates    pkgs/main/win-64::ca-certificates-2020.1.1-0
  certifi            pkgs/main/win-64::certifi-2020.4.5.1-py37_0
  openssl            pkgs/main/win-64::openssl-1.1.1f-he774522_0
  pip                pkgs/main/win-64::pip-20.0.2-py37_1
  python             pkgs/main/win-64::python-3.7.7-h60c2a47_0_cpython
  setuptools         pkgs/main/win-64::setuptools-46.1.3-py37_0
  sqlite             pkgs/main/win-64::sqlite-3.31.1-he774522_0
  vc                 pkgs/main/win-64::vc-14.1-h0510ff6_4
  vs2015_runtime     pkgs/main/win-64::vs2015_runtime-14.16.27012-hf0eaf9b_1
  wheel              pkgs/main/win-64::wheel-0.34.2-py37_0
  wincertstore       pkgs/main/win-64::wincertstore-0.2-py37_0


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

y 입력

```
Downloading and Extracting Packages
certifi-2020.4.5.1   | 156 KB    | #########
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate mytestenv
#
# To deactivate an active environment, use
#
#     $ conda deactivate


(myenv) C:₩Users₩trimu>
```

Anaconda 가상환경 사용법

2) 가상환경 활성화 (activation) / 비활성화 (deactivation)
> conda activate mytestenv
> conda deactivate

3) 가상환경에 설치된 package list 조회
> pip list

4) 필요한 개별 package 설치
```
> conda install pandas
            or
> pip install pandas
```

5) 개별 package 삭제
```
> conda uninstall pandas
            or
> pip uninstall pandas
```

```
(myenv) C:₩Users₩trimu>conda activate mytestenv

(mytestenv) C:₩Users₩trimu>pip list
Package            Version
------------------ -------------------
astroid            2.2.5
certifi            2020.4.5.1
colorama           0.4.1
isort              4.3.21
lazy-object-proxy  1.4.1
mccabe             0.6.1
pip                20.0.2
pylint             2.3.1
setuptools         46.1.3.post20200330
tensorflow-gpu     2.0.0
typed-ast          1.4.0
wheel              0.34.2
wincertstore       0.2
wrapt              1.11.2

(mytestenv) C:₩Users₩trimu>
```

Anaconda 가상환경 사용법

5) 생성되어 있는 가상환경 목록 조회

> conda env list

```
(mytestenv) C:₩Users₩trimu>conda env list
# conda environments:
#
                         C:₩ProgramData₩Anaconda3
my_root                  C:₩Users₩trimu₩AppData₩Local₩conda₩conda₩envs₩my_root
base                     C:₩Users₩trimu₩Miniconda3
flask                    C:₩Users₩trimu₩Miniconda3₩envs₩flask
myenv                    C:₩Users₩trimu₩Miniconda3₩envs₩myenv
mytestenv             *  C:₩Users₩trimu₩Miniconda3₩envs₩mytestenv
py37_32                  C:₩Users₩trimu₩Miniconda3₩envs₩py37_32
tf20                     C:₩Users₩trimu₩Miniconda3₩envs₩tf20
torch                    C:₩Users₩trimu₩Miniconda3₩envs₩torch
```

6) 가상환경 삭제

> conda env remove –n mytestenv

```
(myenv) C:₩Users₩trimu>conda env remove -n mytestenv

Remove all packages in environment C:₩Users₩trimu₩Miniconda3₩envs₩mytestenv:


(myenv) C:₩Users₩trimu>
```

Anaconda 가상환경 사용법

5) 가상환경에서 Python CLI (Command Line Interface) 실행 방법

(myenv) > python

6) Python CLI 종료

(myenv) > exit()

7) 가상환경 prompt 에서 .py 확장자를 가진 프로그램 파일 직접 실행

(myenv) > python myProgram.py

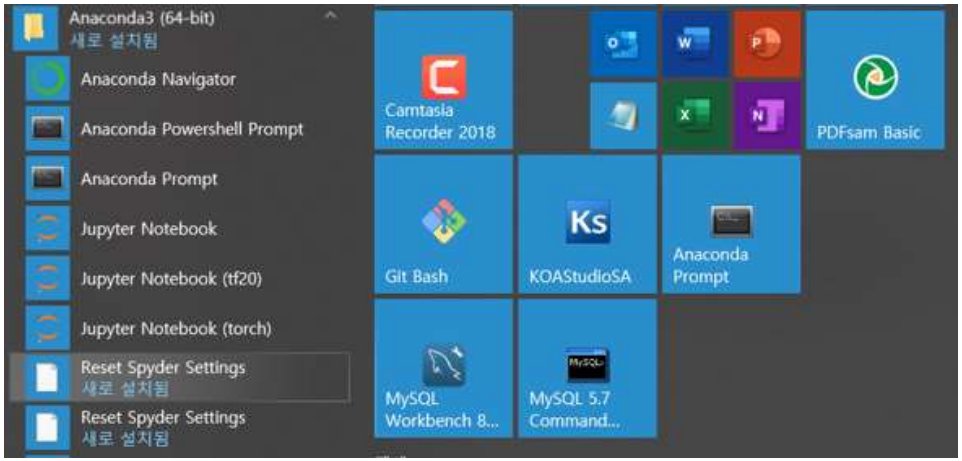
Jupyter Notebook 사용 방법

What is Jupyter Notebook ?

- 주피터 노트북(**Jupyter Notebook**)은 웹 브라우저에서 파이썬 코드를 작성하고 실행해 볼 수 있는 개발도구
- 아나콘다(Anaconda)를 설치하면 **Jupyter Notebook**이 함께 설치
- Notebook 서버 프로그램은 백그라운드에서 실행되는 파이썬 프로그램으로 웹 브라우저를 통해 interactive 하게 Python 명령실행
- 100% open-source software 이고 무료

Jupyter Notebook 실행 방법

1) Start menu 에서 실행



2) 명령 prompt 에서 실행
> jupyter notebook

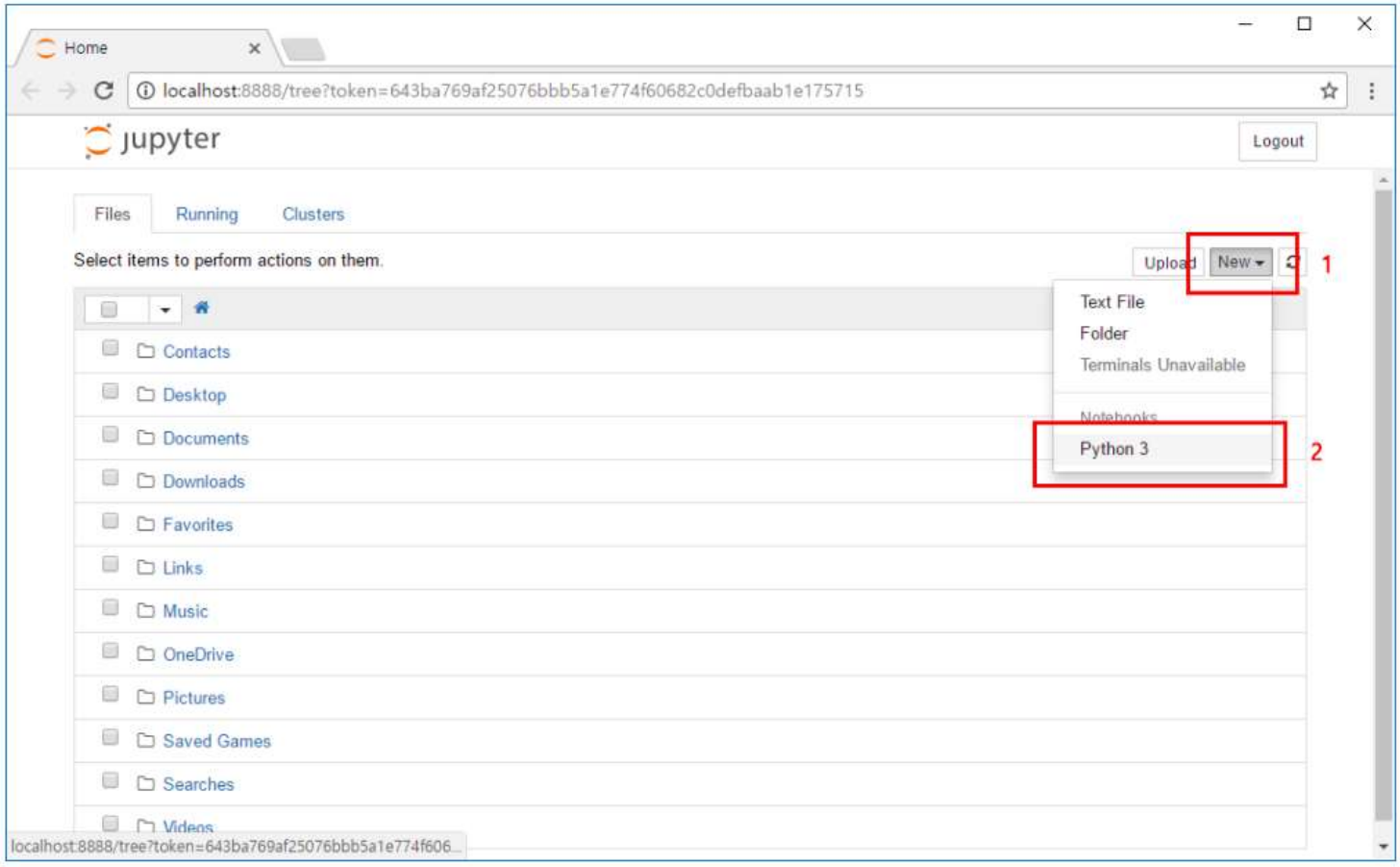
```
(base) C:₩Users₩trimu>jupyter notebook
[I 08:20:31.790 NotebookApp] Serving notebooks from local directory: C:₩Users₩trimu
[I 08:20:31.791 NotebookApp] The Jupyter Notebook is running at:
[I 08:20:31.793 NotebookApp] http://localhost:8888/?token=e1bb3f10db684cd21f9dd3c0e89f713306df271678b97d9d
[I 08:20:31.798 NotebookApp]  or http://127.0.0.1:8888/?token=e1bb3f10db684cd21f9dd3c0e89f713306df271678b97d9d
[I 08:20:31.800 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation
).
[C 08:20:31.994 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///C:/Users/trimu/AppData/Roaming/jupyter/runtime/nbserver-12876-open.html
    Or copy and paste one of these URLs:
        http://localhost:8888/?token=e1bb3f10db684cd21f9dd3c0e89f713306df271678b97d9d
     or http://127.0.0.1:8888/?token=e1bb3f10db684cd21f9dd3c0e89f713306df271678b97d9d
```

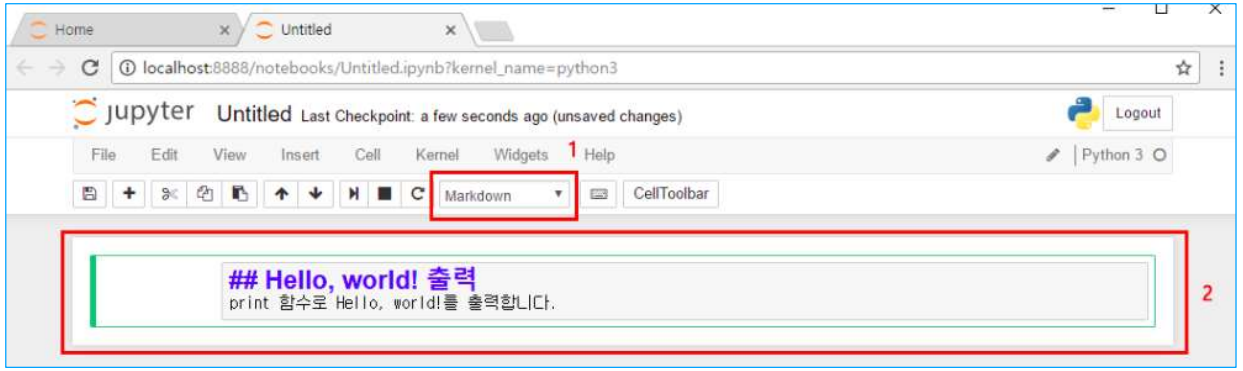
* Login 비밀번호 요구할 경우 사용자 계정의 ~/.jupyter/jupyter_notebook_config.py 삭제

Jupyter Notebook 사용방법

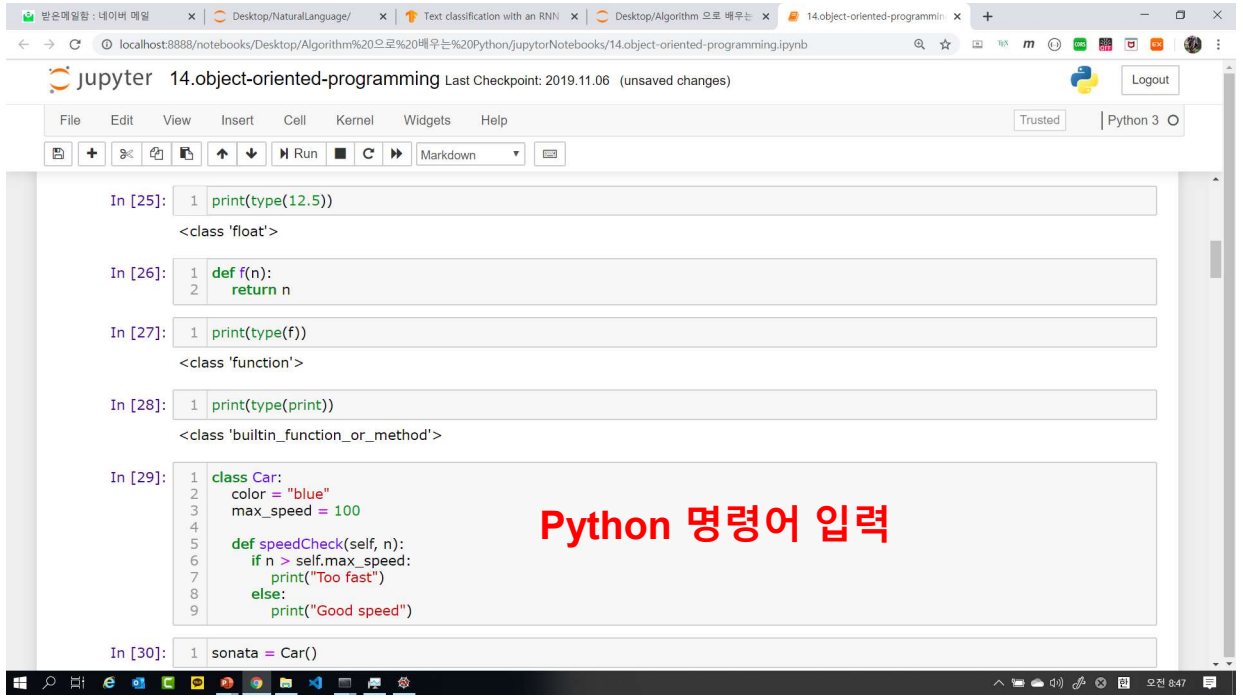
주피터 노트북 초기 화면 에서 New → Python 3 선택 → 새로운 notebook 생성



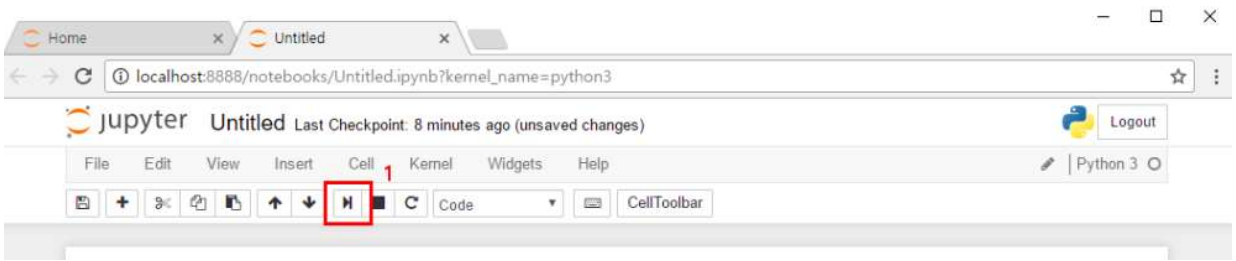
설명 추가



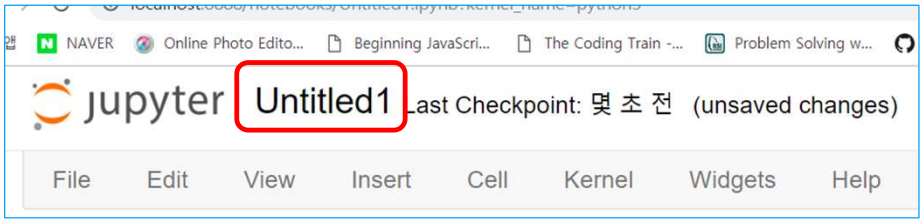
Python Code 입력
및
실행



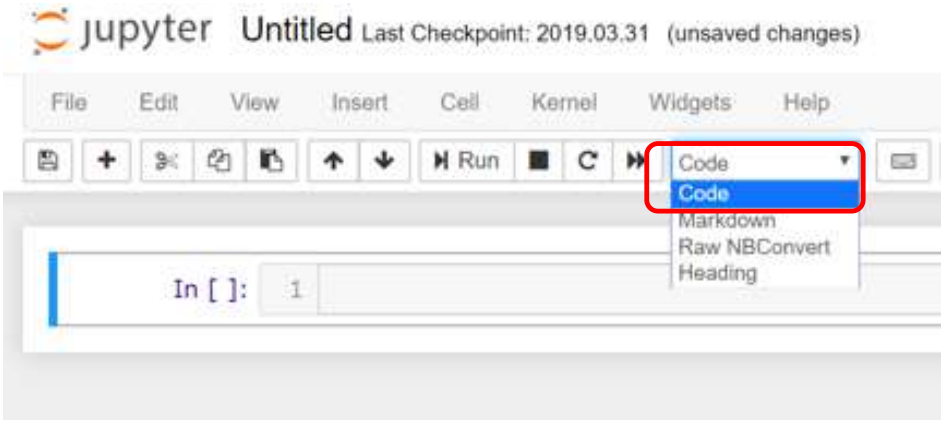
Code 실행



Notebook File Name 바꾸기

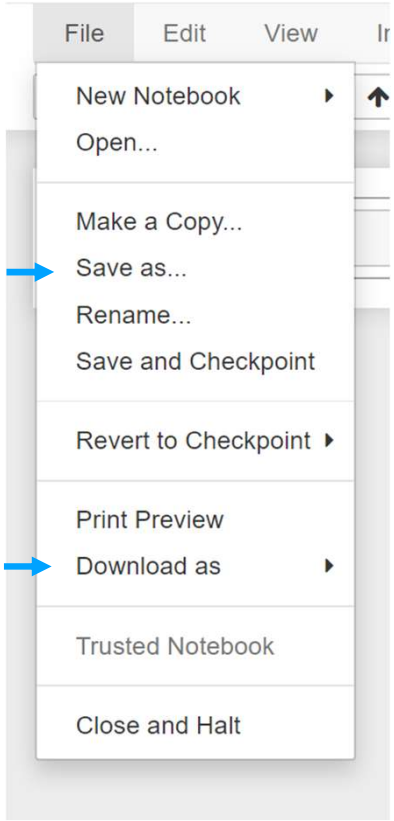


Cell Type 변경



자주 사용하는 menu 들

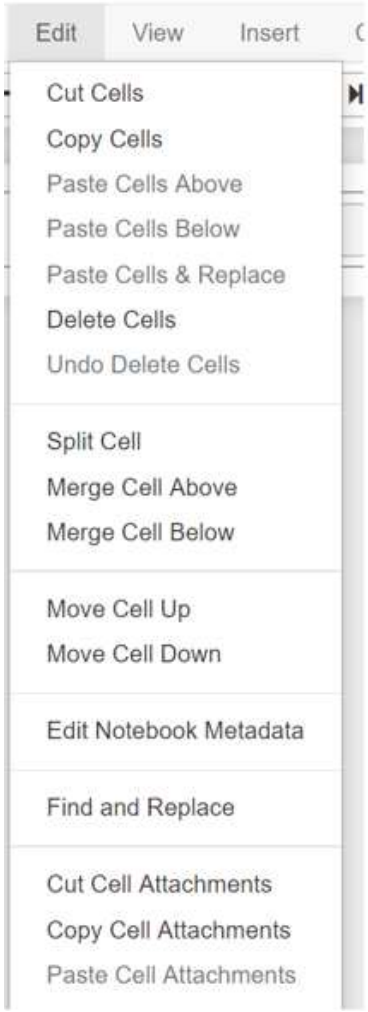
Notebook 저장



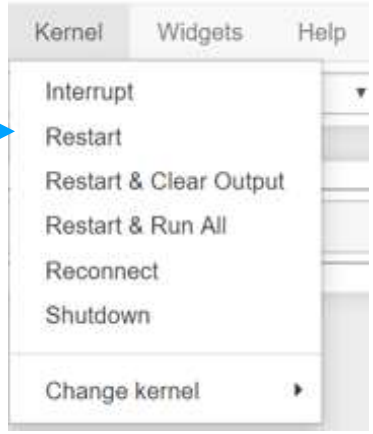
Notebook download

Cell 삭제 취소

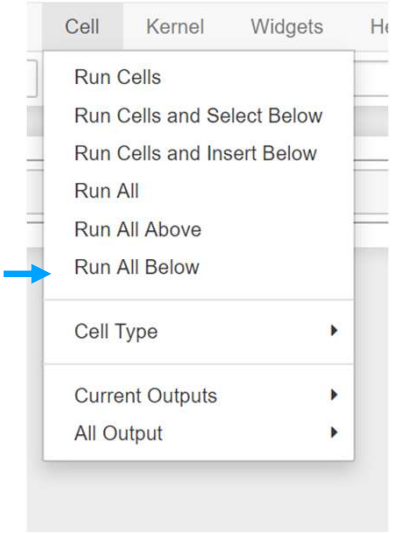
Find/ Replace



Notebook restart



Cell 실행



Jupyter Notebook 사용방법

- 자주 사용하는 short-cut key

Shift + Enter : cell 실행 + 다음 cell 이동

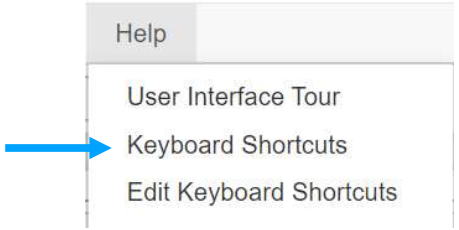
Ctrl + S : save + checkpoint

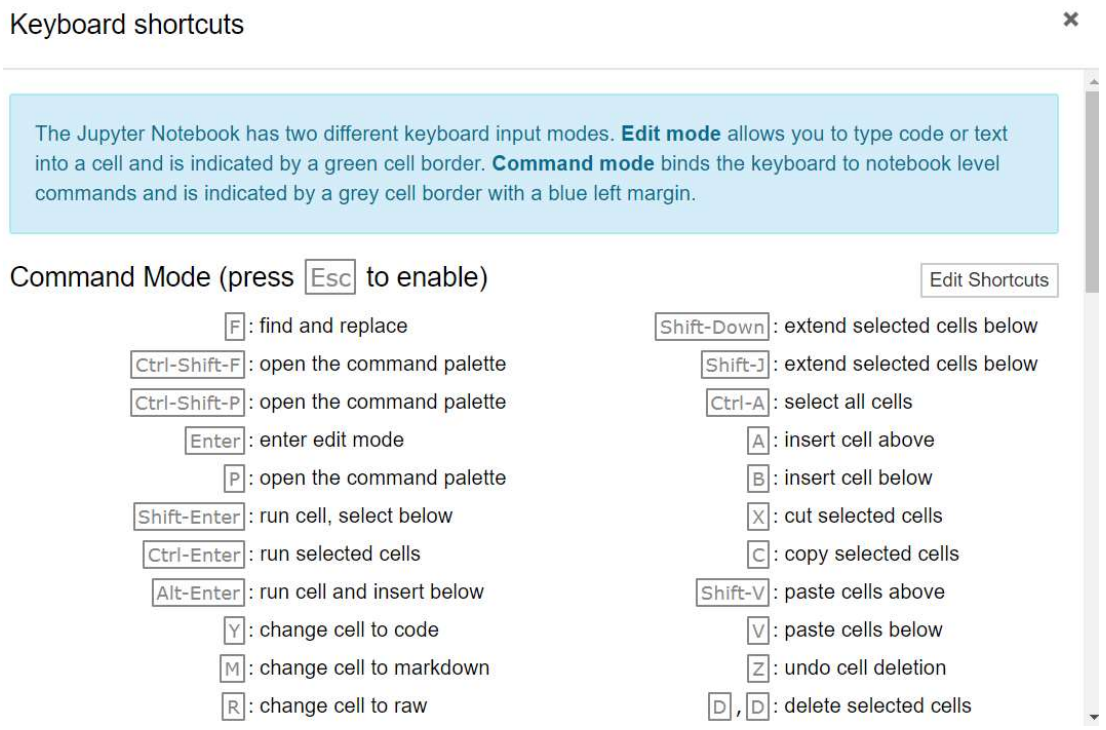
Esc+A : 위쪽에 cell 삽입
Esc+B : 아래쪽에 cell 삽입

Esc+X : cell 삭제

Esc+Z : cell 삭제 취소

Esc+M : cell 을 markdown type 으로 변경



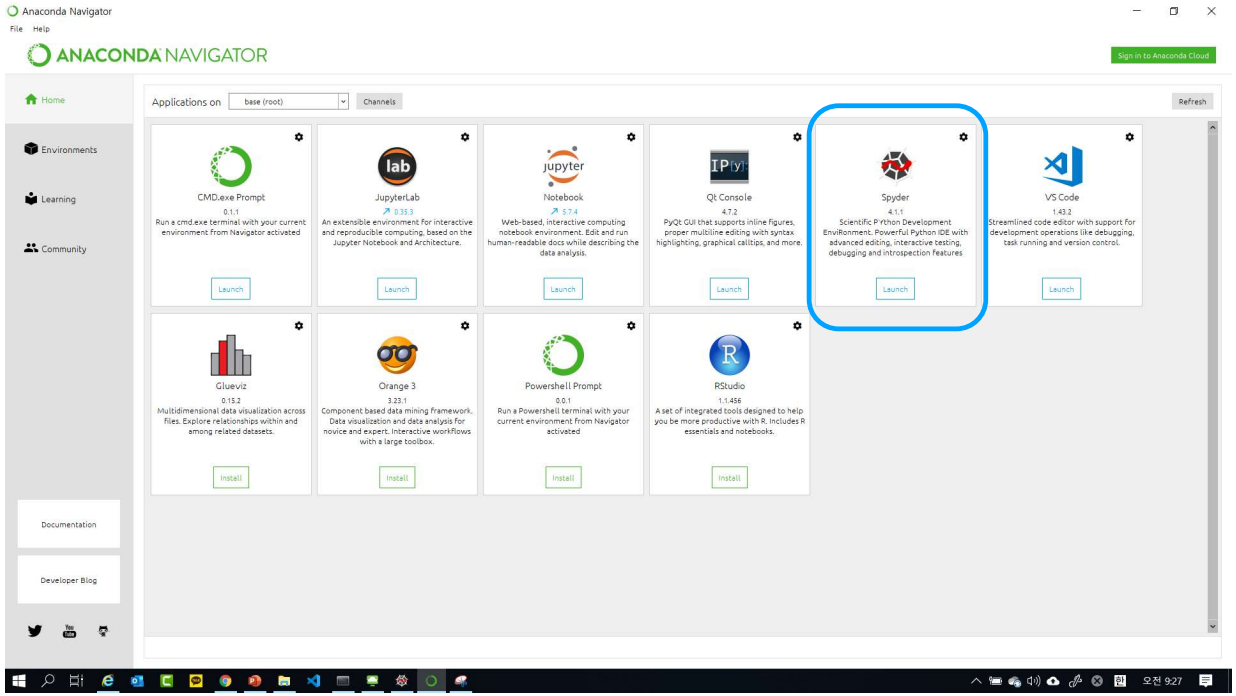


개발도구 사용 방법

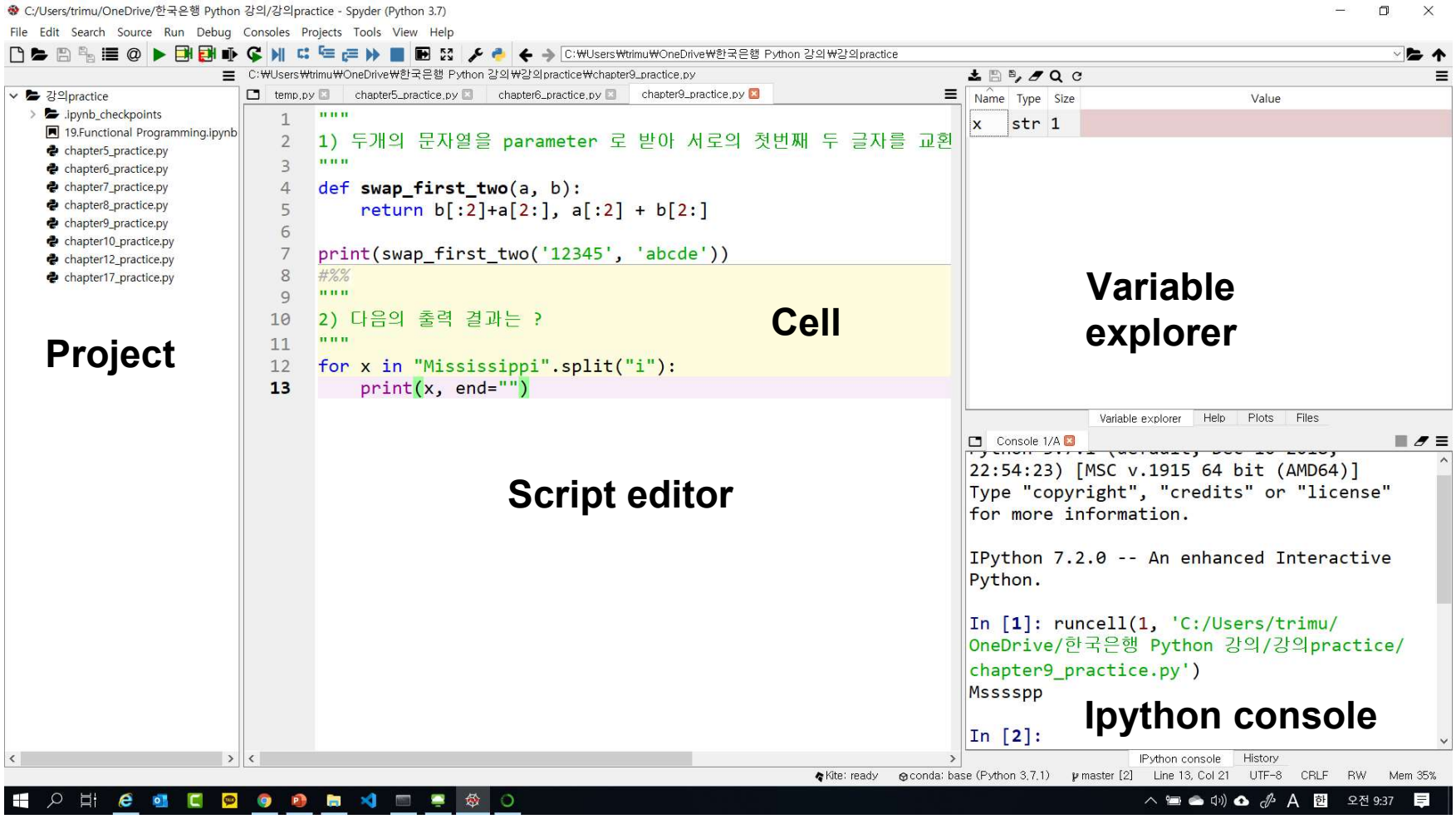
- 개발 도구(Spyder, VS Code) 실행 방법
- Debugging 방법

Spyder

- Code 편집기와 iPython 이 통합된 IDE (Python 에 특화)
- 변수의 내용을 variable explorer 에서 바로 확인
- 일반적으로 VS Code 와 함께 사용하여 생산성 극대화



Spyder 화면 구성

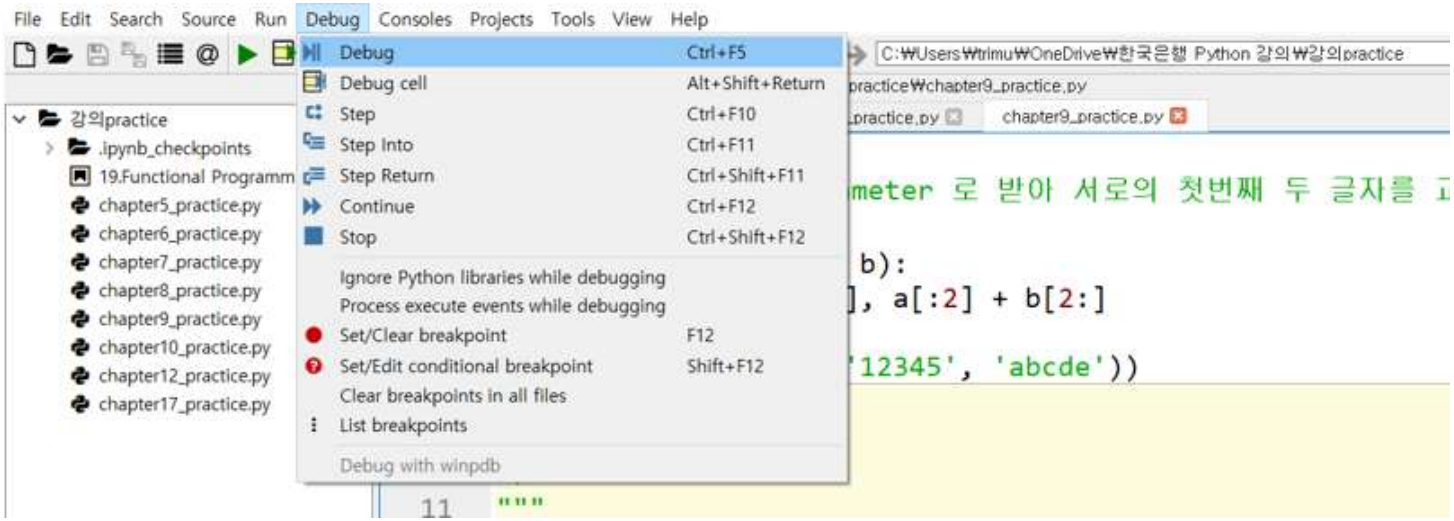

File Edit Search Source Run Debug Consoles Projects Tools View Help
강의practice
.ipynb_checkpoints
19.Functional Programming.ipynb
chapter5_practice.py
chapter6_practice.py
chapter7_practice.py
chapter8_practice.py
chapter9_practice.py
chapter10_practice.py
chapter12_practice.py
chapter17_practice.py
Project
temp.py
chapter5_practice.py
chapter6_practice.py
chapter9_practice.py
"""
1) 두개의 문자열을 parameter 로 받아 서로의 첫번째 두 글자를 교환
"""
def swap_first_two(a, b):
return b[:2]+a[2:], a[:2] + b[2:]
print(swap_first_two('12345', 'abcde'))
#%%
"""
2) 다음의 출력 결과는 ?
"""
for x in "Mississippi".split("i"):
print(x, end="")
Cell
Script editor
Name Type Size Value
x str 1
Variable explorer
Variable explorer Help Plots Files
Console 1/A
22:54:23) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license"
for more information.
IPython 7.2.0 -- An enhanced Interactive Python.
In [1]: runcell(1, 'C:/Users/trimu/OneDrive/한국은행 Python 강의/강의practice/chapter9_practice.py')
Mssspp
In [2]:
Ipython console
IPython console History
Kite: ready
conda: base (Python 3.7.1)
master [2]
Line 13, Col 21
UTF-8
CRLF
RW
Mem 35%
오전 9:37

Spyder code 실행 (run) 방법

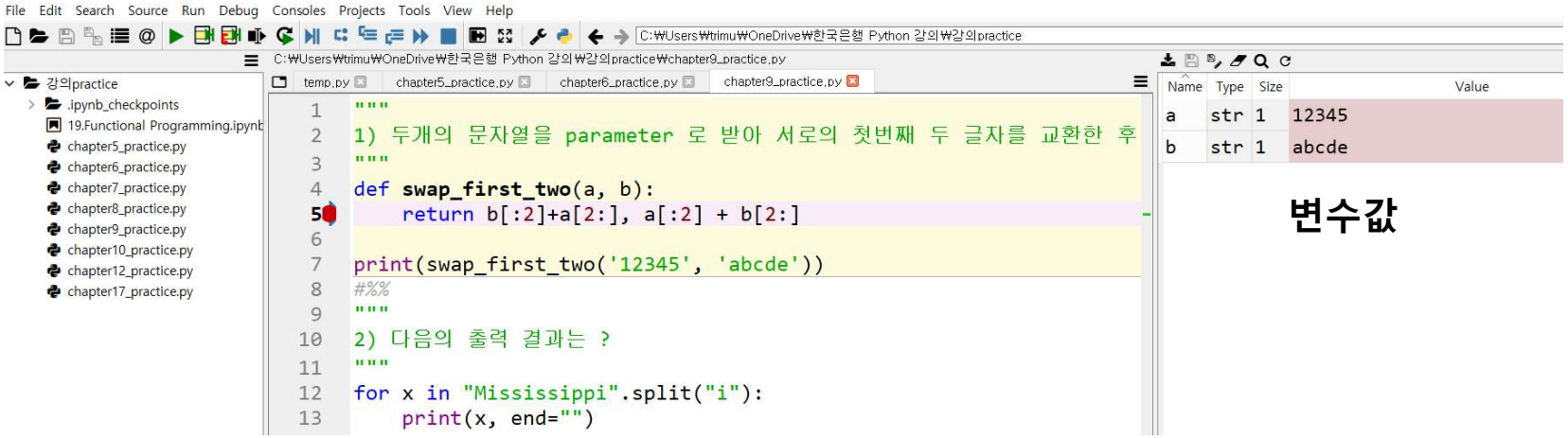
- 파일 전체를 한번에 실행 (Run File) – F5
- 현재 Cell 전체를 실행 (Run Current Cell) – Ctrl+Enter
  - Cell : #%% 로 구분된 code block
- 현재 Cell 전체를 실행하고 다음 Cell 로 cursor 이동 – Shift+Enter
- 각 line 혹은 선택한 lines 실행 (Run Selection) – F9

Spyder debugging

Debug mode 실행



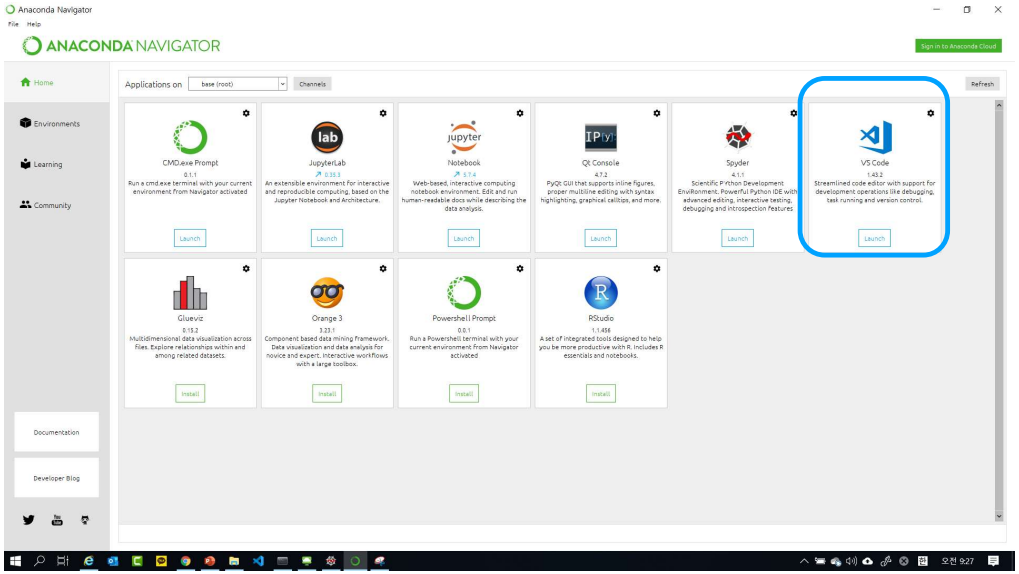
Break Point 설정



변수값

VS Code

- 비주얼 스튜디오 코드는 가볍고, 맥,리눅스,윈도우에서 모두 실행 가능하고, 무료인 코드 편집기.
- 확장 프로그램을 통해 에디터 기능을 확장시킬수 있다.
- 사용설명 (https://demun.github.io/vscode-tutorial/)

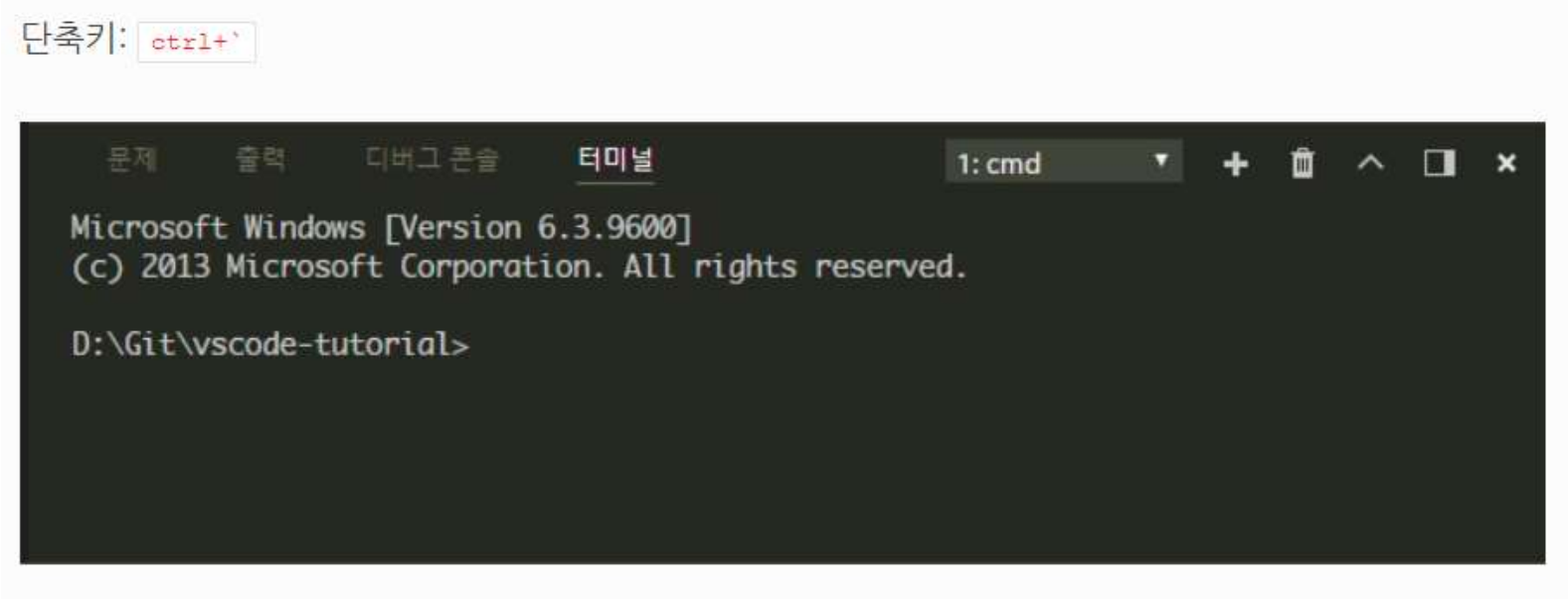


확장프로그램

- 에디터의 기본 기능보다 더 많은 기능을 마켓 플레이스에서 다운로드해서 사용할 수 있다.

터미널

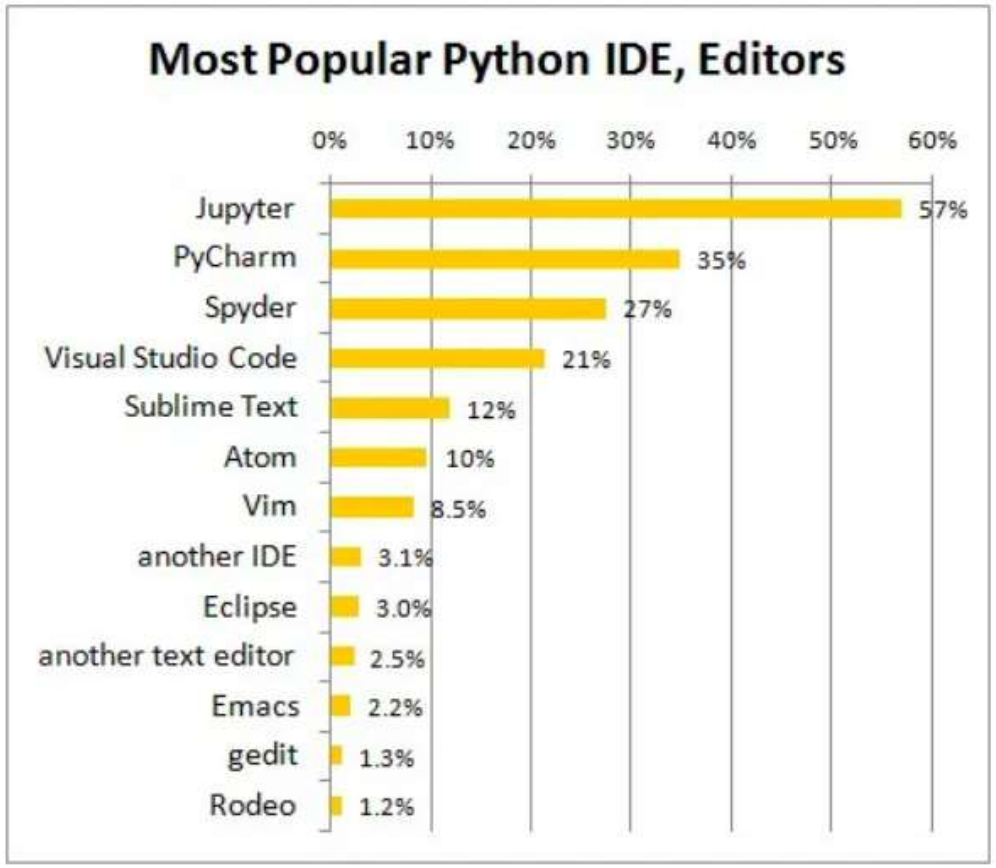
- vscode 는 하단에 터미널을 통해 명령을 실행할 수도 있다.



기타 editors

- PyCharm
- atom
- Sublimetext
- etc

2018. December



Github Repository 를 Local PC 에 복사

https://github.com/ironmanciti/AlgorithmPython