알고리즘으로 배우는 Python

강사: 오영제

Python 의 역사

- 1991 년 Guido Van Rossum 이 개발 – Open Source
- Web programming 에서 Machine Learning 까지 다양한 분야에 사용
- Python 이름은 snake 가 아니라 drama 제목에서 따옴
- Version 2.7 vs Version 3.8

Why Python ?

Python 은 - 배우기 쉽다 - 비교적 빠르다 - Object Oriented Programming - Web 에서 Science 까지 사용 - 이식성이 좋다	- C 는 훨씬 빠르지만 배우기 어렵다 - Java 는 속도와 학습 곡선이 C 와 Python 의 중간

Python vs. Java / C++ 선택시 고려 사항

1. 선택기준 : 프로그램 실행 속도 vs. 개발속도 및 유지보수 용이성

   -. 1970/80 – CPU 속도 및 가격이 language 결정 기준
   -. 프로그램 코드의 가독성 및 유지보수 용이성 – shorter is better

2. Python 의 별명은 glue-language

   -. Python 의 Numpy 모듈은 내부적으로 LPACK 사용
   -. Fortran/C/C++ 라이브러리와 호환
   -. Cython 을 이용하여 Python 을 C code 로 변환 가능

Python 은 대부분 task 에서“fast enough”

*** LPACK (Linear Algebra PACKage with Fortran/C)**

코드길이 비교 – Web Crawler

Java

```
public class CrawlerExample {

    public static void main(String[] args) throws IOException {
        PrintWriter textFile = null;
        try {
            textFile = new PrintWriter("result.txt");
            System.out.println("Enter the URL you wish to crawl..");
            System.out.print("@> ");
            String myUrl = new Scanner(System.in).nextLine();

            String response = getContentByUrl(myUrl);

            Matcher matcher = Pattern
              .compile("href=[\"'](.[^\"']+)[\"']").matcher(response);
            while (matcher.find()) {
                String url = matcher.group(1);
                System.out.println(url);
                textFile.println(url);
            }
        } finally {
            if(textFile != null) {
                textFile.close();
            }
        }
    }

    private static String getContentByUrl(String myUrl)
              throws IOException {
        URL url = new URL(myUrl);
        URLConnection urlConnection = url.openConnection();
        BufferedReader in = null;
        StringBuilder response = new StringBuilder();
        try {
            in = new BufferedReader(new InputStreamReader
              (urlConnection.getInputStream()));
            String inputLine;
            while ((inputLine = in.readLine()) != null) {
                response.append(inputLine);
            }
        } finally {
            if(in != null) {
                in.close();
            }
        }
        return response.toString();
    }
}
```

Python

```
if __name__ == '__main__':
    with open("result.txt","wt") as textFile:
        print("Enter the URL you wish to crawl..")
        myUrl = input("@> ")
        for i in re.findall("href=[\"'](.[^\"']+)[\"']",
            urllib.request.urlopen(myUrl).read().decode(), re.I):
            print(i)
            textFile.write(i+'\n')
```

입력받은 URL Parsing
및 Web Crawling

화면상에서 URL 입력받기

“Hello, World!”

- 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!”)
```

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
```

- 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

- 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 – JSON 과 유사 → {‘cat’: 1, ‘dog’: 2, ‘lion’: 3}

2. 영어 문장과 유사한 명령어 형식

- Java: String name = “Bob”;
  system.out.println(name);

- C++: String name = “Bob”;
  cout << name;

- Python: name = “Bob”
  print(name)

```
for i in range(1, 21):
    print(i)
```

```
for (int i = 1; i < 21; i++){
    system.out.println(i)
}
```

3. One Line Coding (Pythonic Way)

Java:
```
int temp = x;
x = y;
y = temp;
```

Python:
```
x, y = y, x
```

```
lst1 = [ ]
for m in range(4):
    lst2 = [ ]
    for n in range(4):
        lst2.append(n + m*5)
    lst1.append(lst2)
print(lst1)
```

[[n + m*5 for n in range(4)] for m in range(4)]

➔ **[[0, 1, 2, 3], [5, 6, 7, 8], [10, 11, 12, 13], [15, 16, 17, 18]]**

4. Simple Syntax

No { }, No semi-colon / 들여쓰기로 block 구분

```
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

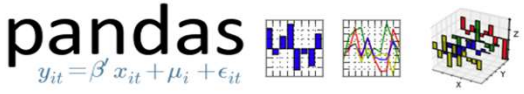
Python: x = 1
x = x / 2 ➔ x : 0.5

6. Python 은 Interpreter 언어

- No compile required

7. 풍부한 Libraries (From Web to Data Science)

Python Libraries



8. Supports both Object Oriented and Functional

OOP (Object Oriented Programming)

Class 를 이용한 객체 생성

객체를 통한 method 호출

Functional Programming

함수 (Function) 정의: 함수A => 함수B => 함수C

입력값 처리
출력값 반환

ex) a = [1, 2, 3, 4, 5], b = [2, 4, 6, 8, 10]

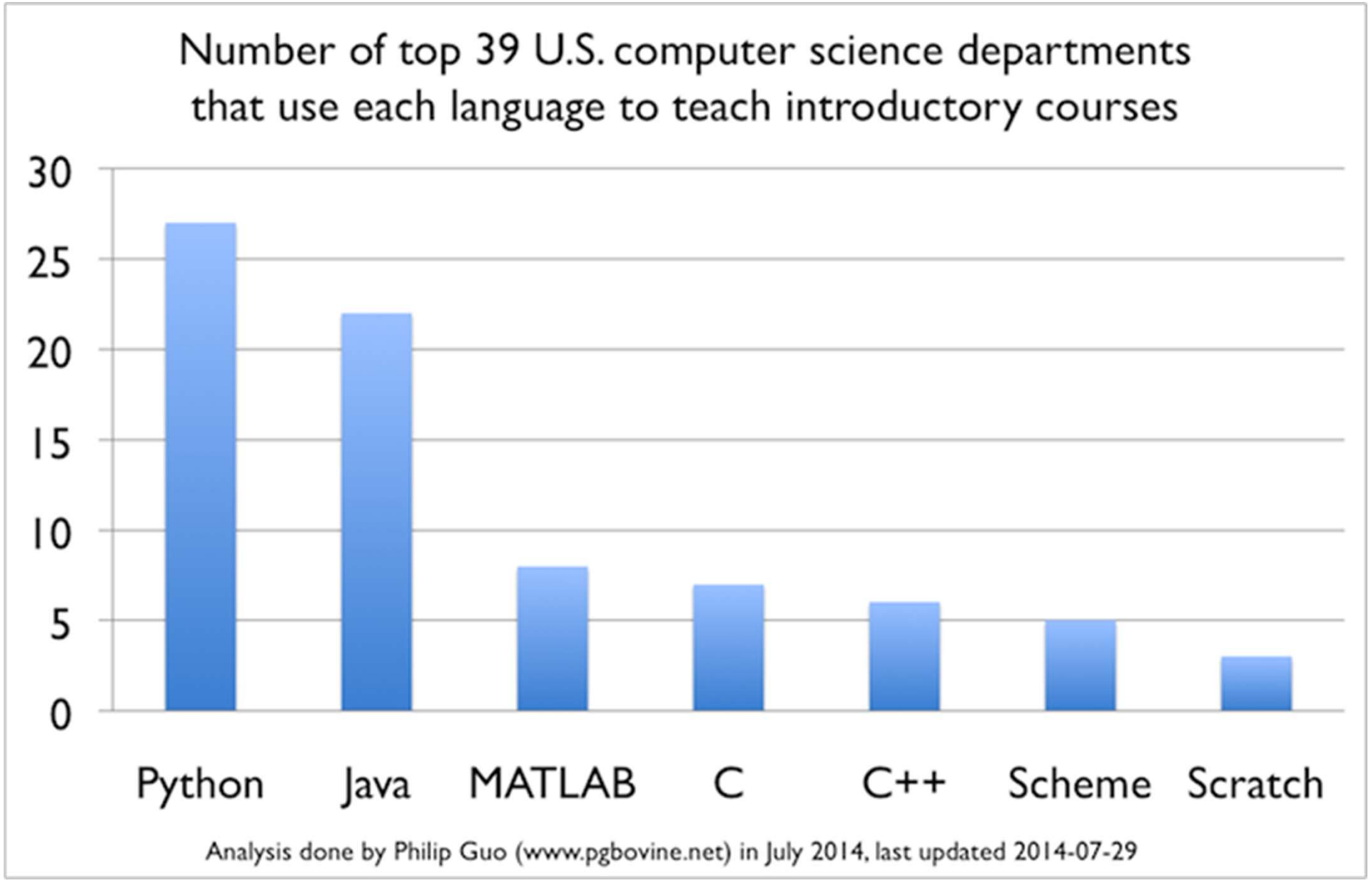
```
list(map(lambda x, y: x * y, a, b))
```

==> [2, 8, 18, 32, 50]

9. Everything Simple coding (다양한 함수 내장)

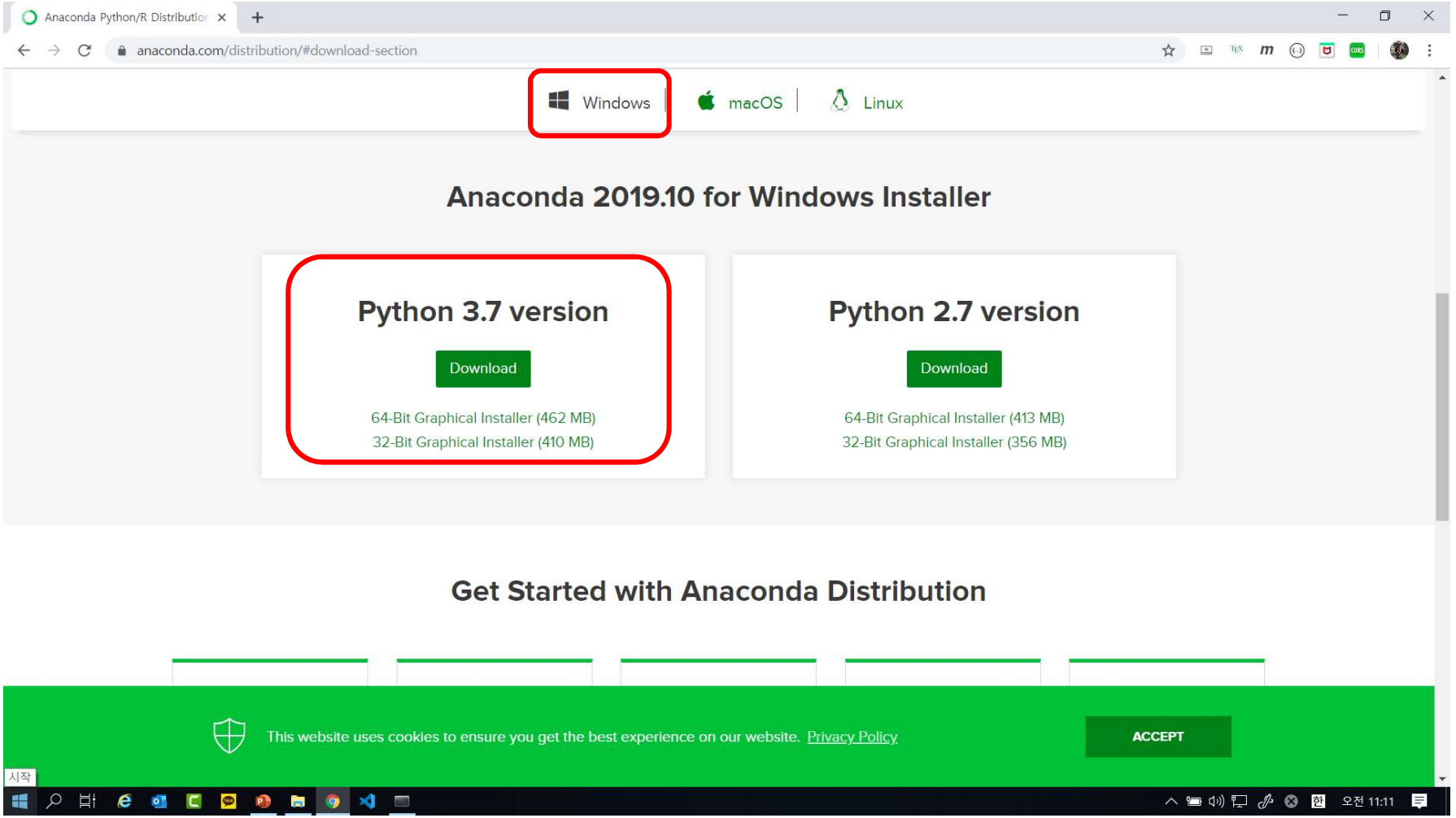
```
>> python –m http-server 8000

>> from collections import counter
>> my_list = [1,1, 2, 3, 4, 3, 2, 3, 4, 2, 1, 2, 3, 5]
>> counter(my_list)
    ➔ counter({2: 4, 1: 3, 3: 4, 4: 2, 5: 1})

>> import itertools
>> itertools.permutation('ab')
    ➔ [('a', 'b'), ('b', 'a')]

>> list = [1, 2, 3, 4, 5]
>> list(reversed(list))
    ➔ [5, 4, 3, 2, 1]
```

Computer Science 학과에서 가르치는 First Language



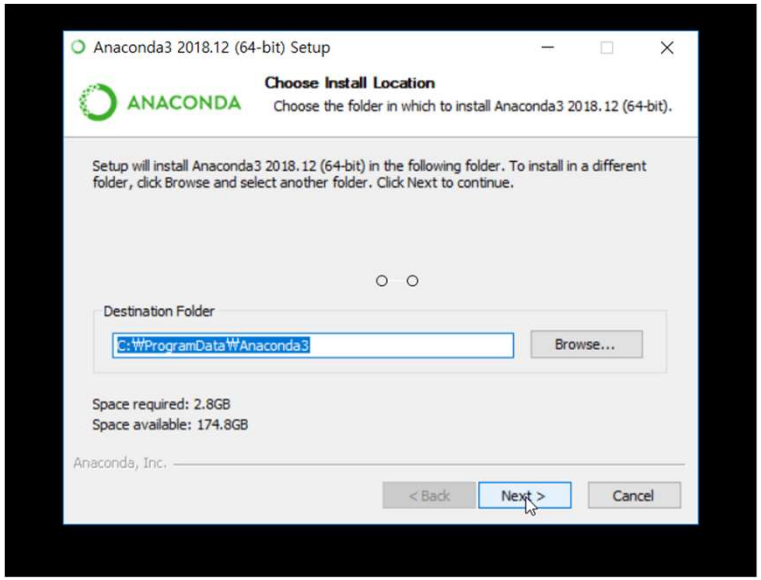
Python 설치 – anaconda (권장)

https://www.anaconda.com/distribution/



설치


Anaconda3 2018.12 (64-bit) Setup
ANACONDA.
Welcome to Anaconda3 2018.12 (64-bit) Setup
Setup will guide you through the installation of Anaconda3 2018.12 (64-bit).
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.
Click Next to continue.
Next >
Cancel


Anaconda3 2018.12 (64-bit) Setup
ANACONDA
Choose Install Location
Choose the folder in which to install Anaconda3 2018.12 (64-bit).
Setup will install Anaconda3 2018.12 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.
Destination Folder
C:₩ProgramData₩Anaconda3
Browse...
Space required: 2.8GB
Space available: 174.8GB
Anaconda, Inc.
< Back
Next >
Cancel

Python package 설치 (pip)

- 작업 directory 생성: mkdir xxxx
- 가상환경 생성 (필요시)
  conda create –name xxxxx python=3.7
  conda activate xxxxx
  conda deactivate
- 패키지 설치 : pip install package-name
- 패키지 제거 : pip uninstall package-name
- 최신버전 updgrade : pip install –upgrade package-name
- 설치된 Python package 확인 : pip list

Python 실행

- Anaconda 명령 prompt

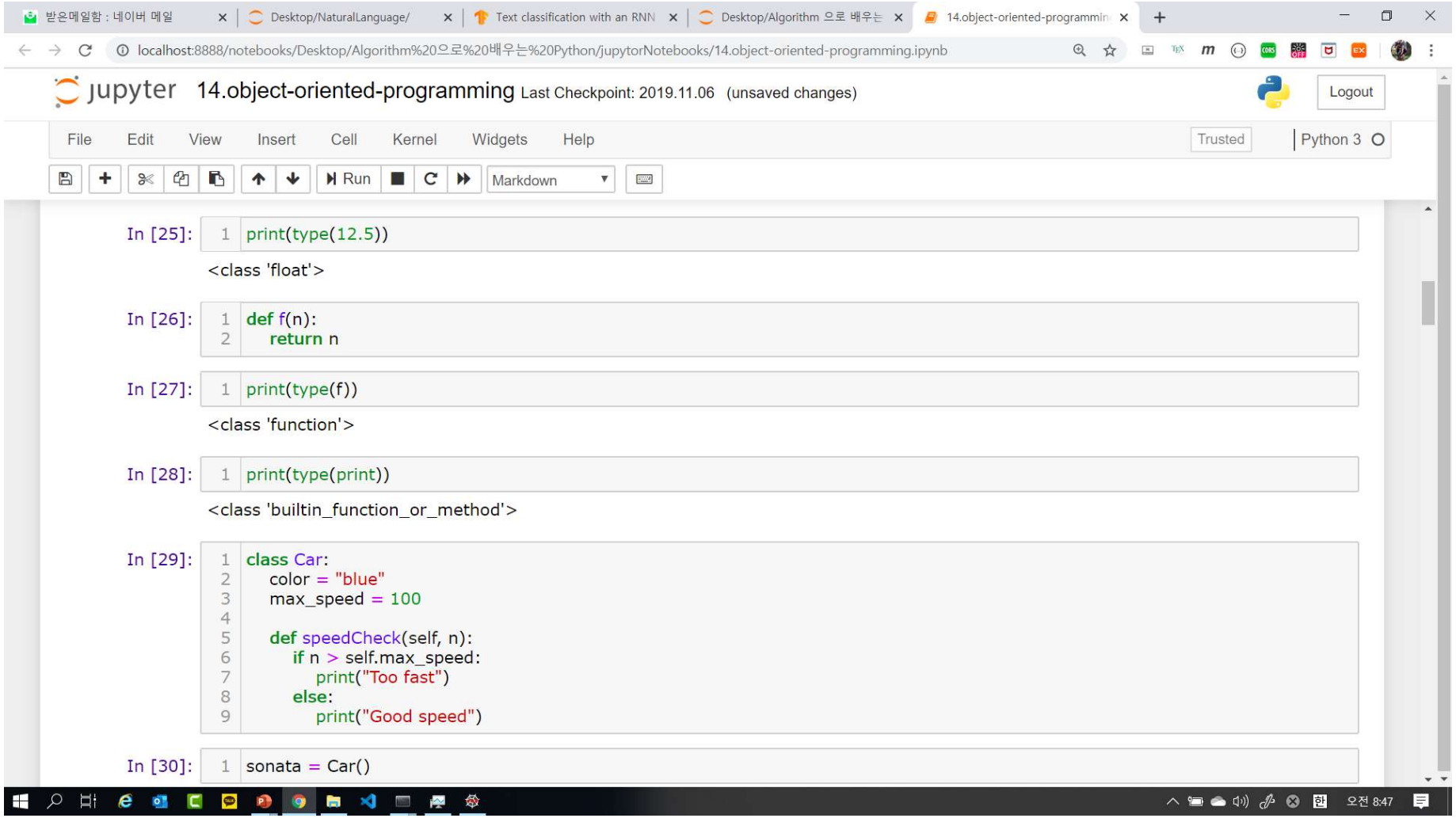
(base) C:\Users\trimu>python

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

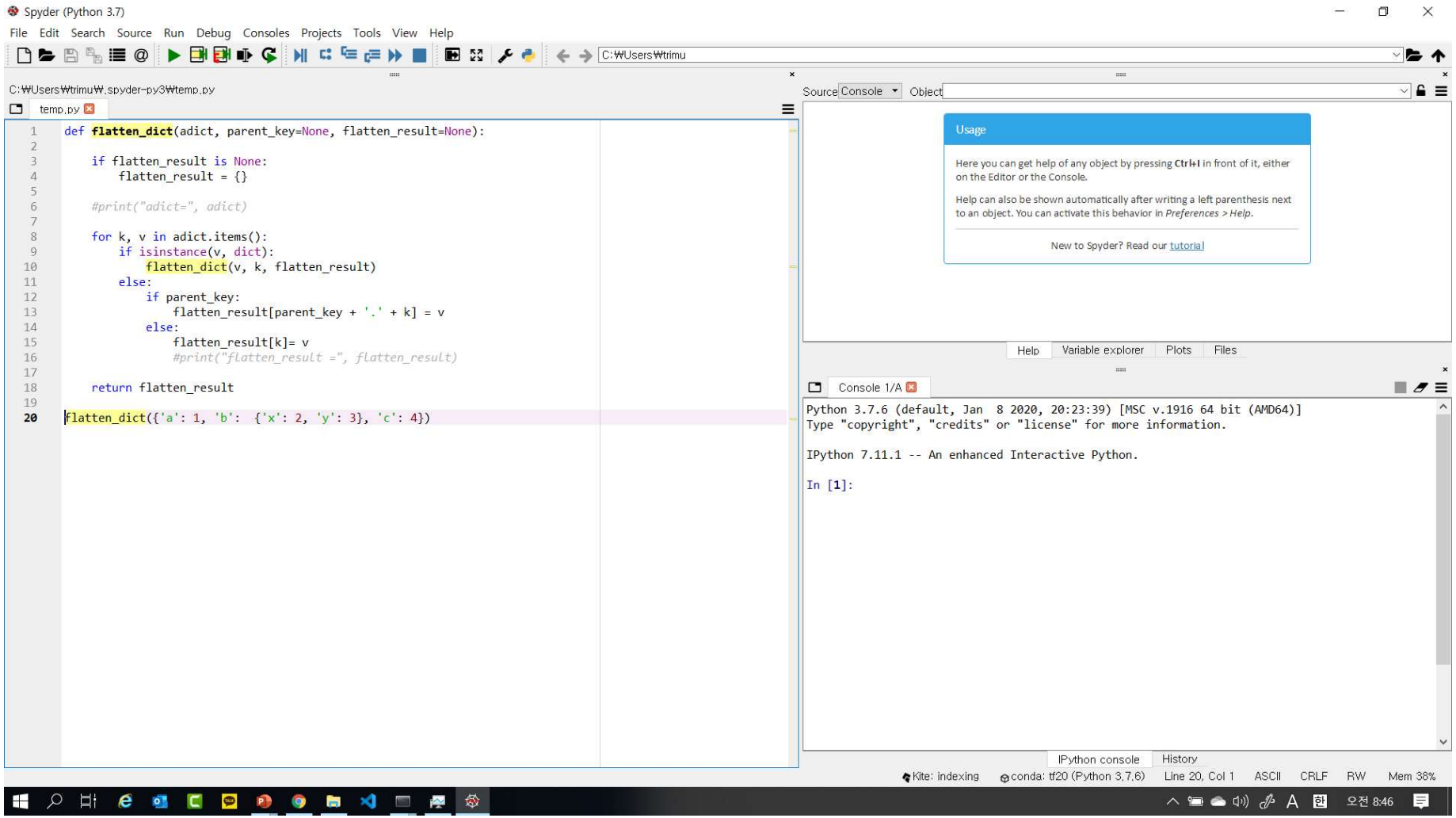
Type "help", "copyright", "credits" or "license" for more information.

>>>

Jupyter Notebook 에서 Python 수행



SPYDER


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:₩Users₩trimu
C:₩Users₩trimu₩.spyder-py3₩temp.py
temp.py
def flatten_dict(adict, parent_key=None, flatten_result=None):
if flatten_result is None:
flatten_result = {}
#print("adict=", adict)
for k, v in adict.items():
if isinstance(v, dict):
flatten_dict(v, k, flatten_result)
else:
if parent_key:
flatten_result[parent_key + '.' + k] = v
else:
flatten_result[k]= v
#print("flatten_result =", flatten_result)
return flatten_result
flatten_dict({'a': 1, 'b': {'x': 2, 'y': 3}, 'c': 4})
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help Variable explorer Plots Files
Console 1/A
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.11.1 -- An enhanced Interactive Python.
In [1]:
IPython console History
Kite: indexing
conda: tf20 (Python 3.7.6)
Line 20, Col 1
ASCII
CRLF
RW
Mem 38%
오전 8:46

기타

- PyCharm
- atom
- Sublimetext
- Visual Studio Code, etc

Github Repository

https://github.com/ironmanciti/algorithmPython