XML Overview

XML and JSON

최성철 교수
Director of TEAMLAB

우리가 처리하는
데이터 저장 방식들

CSV, HTML

XML, JSON

eXtensible Markup Language

XML이란

- 데이터의 구조와 의미를 설명하는
  TAG(MarkUp)를 사용하여 표시하는 언어

- TAG와 TAG사이에 값이 표시되고,
  구조적인 정보를 표현할 수 있음

- HTML과 문법이 비슷, 대표적인 데이터 저장 방식

XML이란

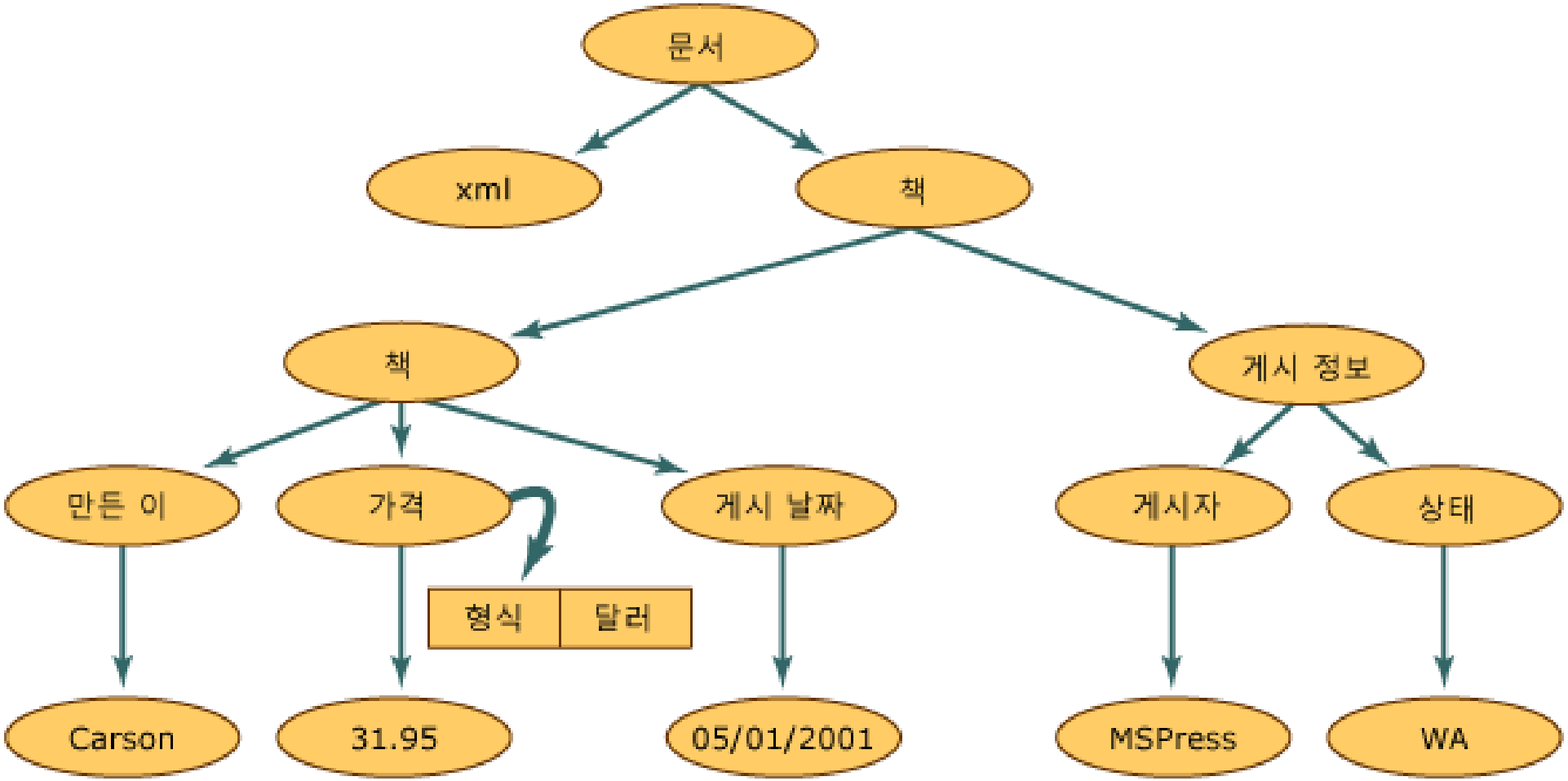
- 정보의 구조에 대한 정보인 스키마와 DTD 등으로 정보에 대한 정보(메타정보)가 표현되며, 용도에 따라 다양한 형태로 변경가능

- XML은 컴퓨터(예: PC ↔ 스마트폰)간에 정보를 주고받기 매우 유용한 저장 방식으로 쓰이고 있음

XML 예제

```
<?xml version="1.0"?>
<고양이>
  <이름>나비</이름>
  <품종>샴</품종>
  <나이>6</나이>
  <중성화>예</중성화>
  <발톱 제거>아니요</발톱 제거>
  <등록 번호>lzz138bod</등록 번호>
  <소유자>이강주</소유자>
</고양이>
```

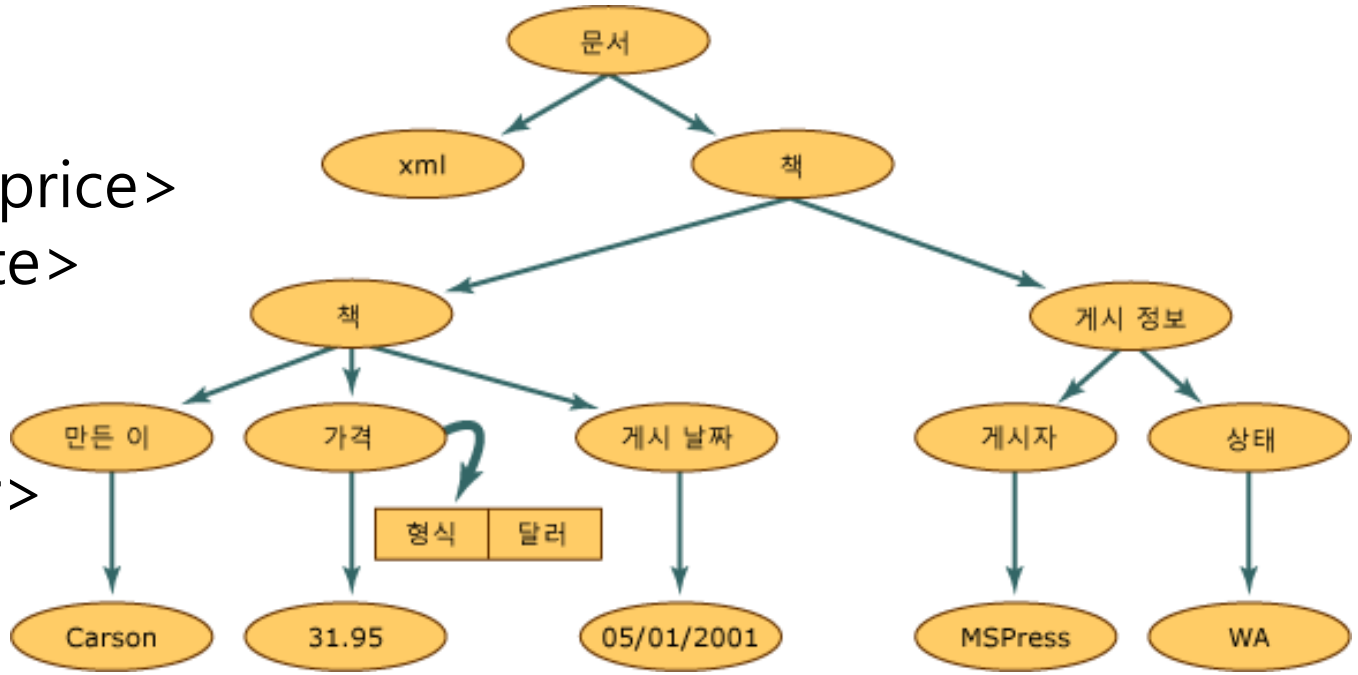
XML 형태로 만들어 보기



http://goo.gl/7mO15w

XML 형태로 만들어 보기

```
<?xml version="1.0"?>
  <books>
    <book>
        <author>Carson</author>
        <price format="dollar">31.95</price>
        <pubdate>05/01/2001</pubdate>
    </book>
    <pubinfo>
        <publisher>MSPress</publisher>
        <state>WA</state>
    </pubinfo>
  </books>
```



http://goo.gl/7mO15w

XML Parsing in Python

- XML도 HTML과 같이 구조적 MarkUp 언어

- 정규표현식으로 Parsing이 가능함

- 그러나 좀 더 손쉬운 도구들이 개발되어 있음

- 가장 많이 쓰이는 parser인 beautifulsoup으로 파싱

Human knowledge belongs to the world.

Lab – XML Parsing

Web Handling

최성철 교수
Director of TEAMLAB

BeautifulSoup

- HTML, XML등 Markup 언어 Scraping을 위한 대표적인 도구
- https://www.crummy.com/software/BeautifulSoup/
- lxml 과 html5lib 과 같은 Parser를 사용함
- 속도는 상대적으로 느리나 간편히 사용할 수 있음

https://www.crummy.com/software/BeautifulSoup/bs4/doc/

BeautifulSoup

	Python 2.7		Python 3.2	
Parser	Speed (KB/s)	Success rate	Speed (KB/s)	Success rate
Beautiful Soup 3.2 (SGMLParser)	211	100%	-	-
html5lib (BS3 treebuilder)	253	99%	-	-
Beautiful Soup 4.0 + lxml	255	100%	2140	96%
html5lib (lxml treebuilder)	270	99%	-	-
Beautiful Soup 4.0 + html5lib	271	98%	-	-
Beautiful Soup 4.0 + HTMLParser	299	59%	1705	57%
html5lib (simpletree treebuilder)	332	100%	-	-
HTMLParser	5194	52%	3918	57%
lxml	17925	100%	14258	96%

https://www.crummy.com/2012/01/22/0

beautifulsoup 설치

- conda 가상 환경으로 lxml과 beautifulsoup 설치

```
activate python_mooc
conda install lxml
conda install -c anaconda beautifulsoup4=4.5.1
```

beautifulsoup 모듈 사용

- 모듈 호출

```
from bs4 import BeautifulSoup
```

- 객체 생성

```
soup  = BeautifulSoup(books_xml, "lxml")
```

- Tag 찾는 함수 find_all 생성

```
soup.find_all("author")
```

beautifulsoup 모듈 사용

- **find_all: 정규식과 마찬가지로 해당 패턴을 모두 반환**
- **find('invention-title')**

Tag 네임 = title

- **get_text(): 반환된 패턴의 값 반환 (태그와 태그 사이)**

<invention-title id="d2e43">
Adjustable shoulder device for hard upper torso suit
</invention-title>

http://goo.gl/aeKMGS, http://goo.gl/lKhFzh 참고

beautifulsoup Example

- 데이터 다운로드 받기

https://s3.ap-northeast-2.amazonaws.com/teamlab-gachon/books.xml

```
from bs4 import BeautifulSoup

with open("books.xml", "r", encoding="utf8") as books_file:
    books_xml = books_file.read()  # File을 String으로 읽어오기

soup  = BeautifulSoup(books_xml, "lxml") # lxml Parser를 사용해서 데이터 분석

# author가 들어간 모든 element 추출
for book_info in soup.find_all("author"):
    print (book_info)
    print (book_info.get_text())
```

beautifulsoup 예제 데이터

- 미국 특허청 (USPTO) 특허 데이터는 XML로 제공됨

- 해당 데이터중 등록번호 “08621662” 인

  “Adjustable shoulder device for hard upper torso suit” 분석

  참고: http://www.google.com/patents/US20120260387

  https://s3.ap-northeast-2.amazonaws.com/teamlab-gachon/US08621662-20140107.XML

- XML 데이터를 Beautiful Soup을 통해 데이터 추출

beautifulsoup 예제 데이터

- 데이터 다운로드 받기

https://s3.ap-northeast-2.amazonaws.com/teamlab-gachon/US08621662-20140107.XML

```
import urllib.request
from bs4 import BeautifulSoup

with open("US08621662-20140107.XML", "r", encoding="utf8") as patent_xml:
    xml = patent_xml.read()  # File을 String으로 읽어오기

soup = BeautifulSoup(xml, "lxml") #lxml parser 호출

#invention-title tag 찾기
invention_title_tag = soup.find("invention-title")
print (invention_title_tag.get_text())
```

beautifulsoup 예제 데이터 응용

- 특허의 출원번호, 출원일, 등록번호, 등록일, 상태, 특허명을 추출

```
<publication-reference>        등록 관련 정보
<document-id>
<country>US</country>
<doc-number>08621662</doc-number> 등록번호
<kind>B2</kind>    상태
<date>20140107</date>        등록일자
</document-id>
</publication-reference>

<application-reference appl-type="utility"> 출원 관련 정보
<document-id>
<country>US</country>
<doc-number>13175987</doc-number> 출원 번호
<date>20110705</date>        출원일
</document-id>
</application-reference>
```

beautifulsoup 예제 데이터 응용

```
publication_reference_tag = soup.find("publication-reference")
p_document_id_tag = publication_reference_tag.find("document-id")
p_country = p_document_id_tag.find("country").get_text()
p_doc_number = p_document_id_tag.find("doc-number").get_text()
p_kind = p_document_id_tag.find("kind").get_text()
p_date = p_document_id_tag.find("date").get_text()
```

```
application_reference_tag = soup.find("application-reference")
a_document_id_tag = publication_reference_tag.find("document-id")
a_country = p_document_id_tag.find("country").get_text()
a_doc_number = p_document_id_tag.find("doc-number").get_text()
a_date = p_document_id_tag.find("date").get_text()
```

```
<publication-reference>
<document-id>
<country>US</country>
<doc-number>08621662</doc-number>
<kind>B2</kind>
<date>20140107</date>
</document-id>
</publication-reference>

<application-reference appl-type="utility">
<document-id>
<country>US</country>
<doc-number>13175987</doc-number>
<date>20110705</date>
</document-id>
</application-reference>
```

[연습] ipg140107.xml 분석

- **ipa110106.xml 파일은 11년 첫째주에 나온 출원 특허를 모은 파일**

  https://s3.ap-northeast-2.amazonaws.com/teamlab-gachon/ipa110106.XML

- **개별 특허들을 나눠서 CSV 형태로 저장 하는 문제**

- **개별 특허 시작은 *<?xml version="1.0"* 시작함**

- **분할된 특허 문서로 부터 특허의 등록번호, 등록일자, 출원 번호, 출원 일자, 상태, 특허 제목을 추출하여 CSV로 만들 것**

Human knowledge belongs to the world.

JSON Overview

XML and JSON

최성철 교수
Director of TEAMLAB

JavaScript Object Notation

JSON

- JavaScript Object Notation

- 원래 웹 언어인 Java Script의 데이터 객체 표현 방식

- 간결성으로 기계/인간이 모두 이해하기 편함

- 데이터 용량이 적고, Code로의 전환이 쉬움

- 이로 인해 XML의 대체제로 많이 활용되고 있음

JSON 예시

```
{
  "users": [
    {
      "name": "John",
      "age": 25
    },
    {
      "name": "Mark",
      "age": 29
    },
    {
      "name": "Sarah",
      "age": 22
    }
  ],
  "dataTitle": "JSON Tutorial!",
  "swiftVersion": 2.1
}
```

Python의 Dict Type과 유사,

Key:Value 쌍으로 데이터 표시

https://goo.gl/gVy0Ms

JSON vs XML

http://www.pcmag.com/encyclopedia/term/56790/json

JSON

```
{
"siblings": [
{"firstName":"Anna","lastName":"Clayton"},
{"lastName":"Alex","lastName":"Clayton"}
]
}
```

XML

```
<siblings>
<sibling>
<firstName>Anna</firstName>
<lastName>Clayton</lastName>
</sibling>
<sibling>
<firstName>Alex</firstName>
<lastName>Clayton</lastName>
</sibling>
</siblings>
```

JSON in Python

- json 모듈을 사용하여 손 쉽게 파싱 및 저장 가능
- 데이터 저장 및 읽기는 dict type과 상호 호환 가능
- 웹에서 제공하는 API는 대부분 정보 교환 시 JSON 활용
- 페이스북, 트위터, Github 등 거의 모든 사이트
- 각 사이트 마다 Developer API의 활용법을 찾아 사용

Human knowledge belongs to the world.

Lab – JSON Handling

Web Handling

최성철 교수
Director of TEAMLAB

JSON Read

- JSON 파일의 구조를 확인 → 읽어온 후 → Dict Type처럼 처리

```
{"employees":[
    {"firstName":"John", "lastName":"Doe"},
    {"firstName":"Anna", "lastName":"Smith"},
    {"firstName":"Peter", "lastName":"Jones"}
]}
```

JSON Data

https://s3.ap-northeast-2.amazonaws.com/teamlab-gachon/json_example.json

```
import json

with open("json_example.json", "r", encoding="utf8") as f:
    contents = f.read()
    json_data =  json.loads(contents)
    print(json_data["employees"])
```

JSON Write

- Dict Type으로 데이터 저장 → josn모듈로 Write

```
import json

dict_data = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}

with open("data.json", "w") as f:
    json.dump(dict_data, f)
```

XML Read → JSON Write

- CSV로 저장된 ipa110106.xml 파일을 JSON으로 변환하기

  https://s3.ap-northeast-2.amazonaws.com/teamlab-gachon/ipa110106.XML

- 분할된 특허 문서로 부터 특허의 등록번호, 등록일자, 출원 번호, 출원 일자, 상태, 특허 제목을 추출하여 JSON로 만들 것

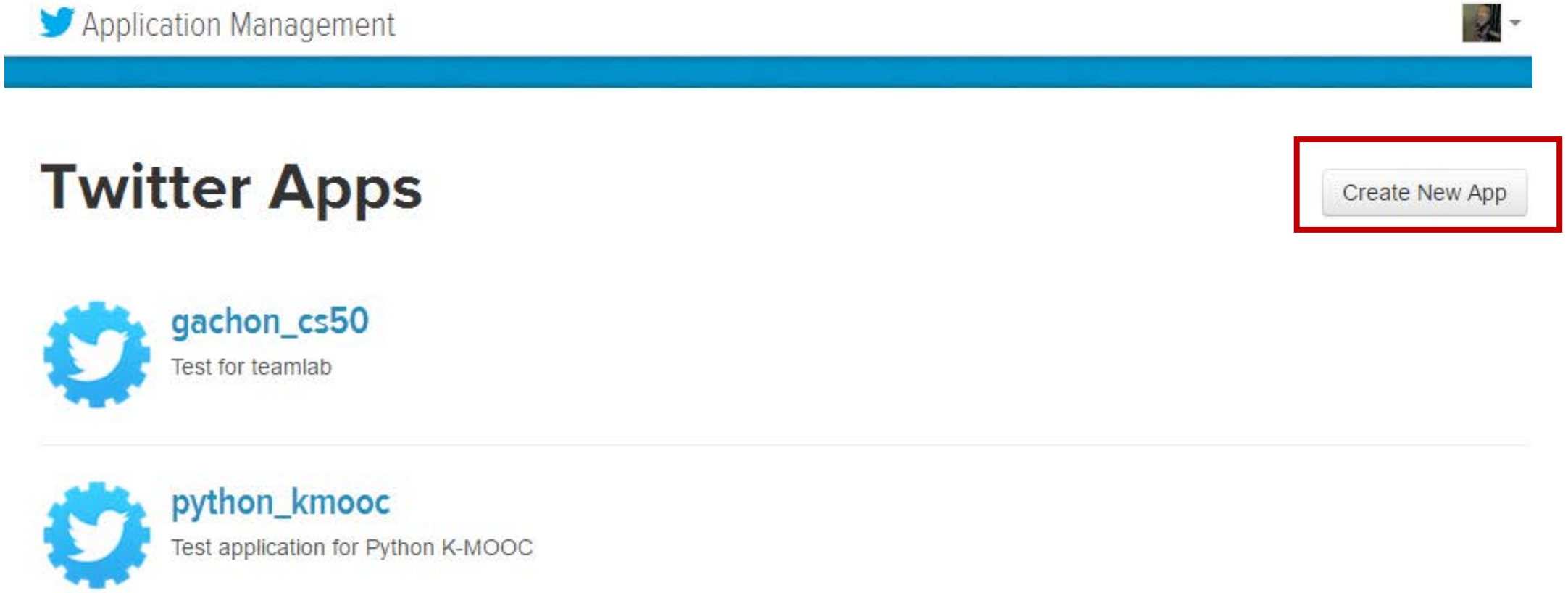
- 각 문서의 Key 값으로 Application ID를 활용함

Twitter 데이터 가져오기 http://jinse.datastats.info/1

- Twitter에서 제공하는 Developer API를 사용하여 트위터 데이터 수집
- 수집되는 데이터 형태는 JSON 형태로 제공함
- https://dev.twitter.com/ Oauth 인증으로 데이터를 주고 받을 수 있음
- 다양한 기능을 이해하기 위해 API 문서의 공부가 필요
  https://dev.twitter.com/overview/api

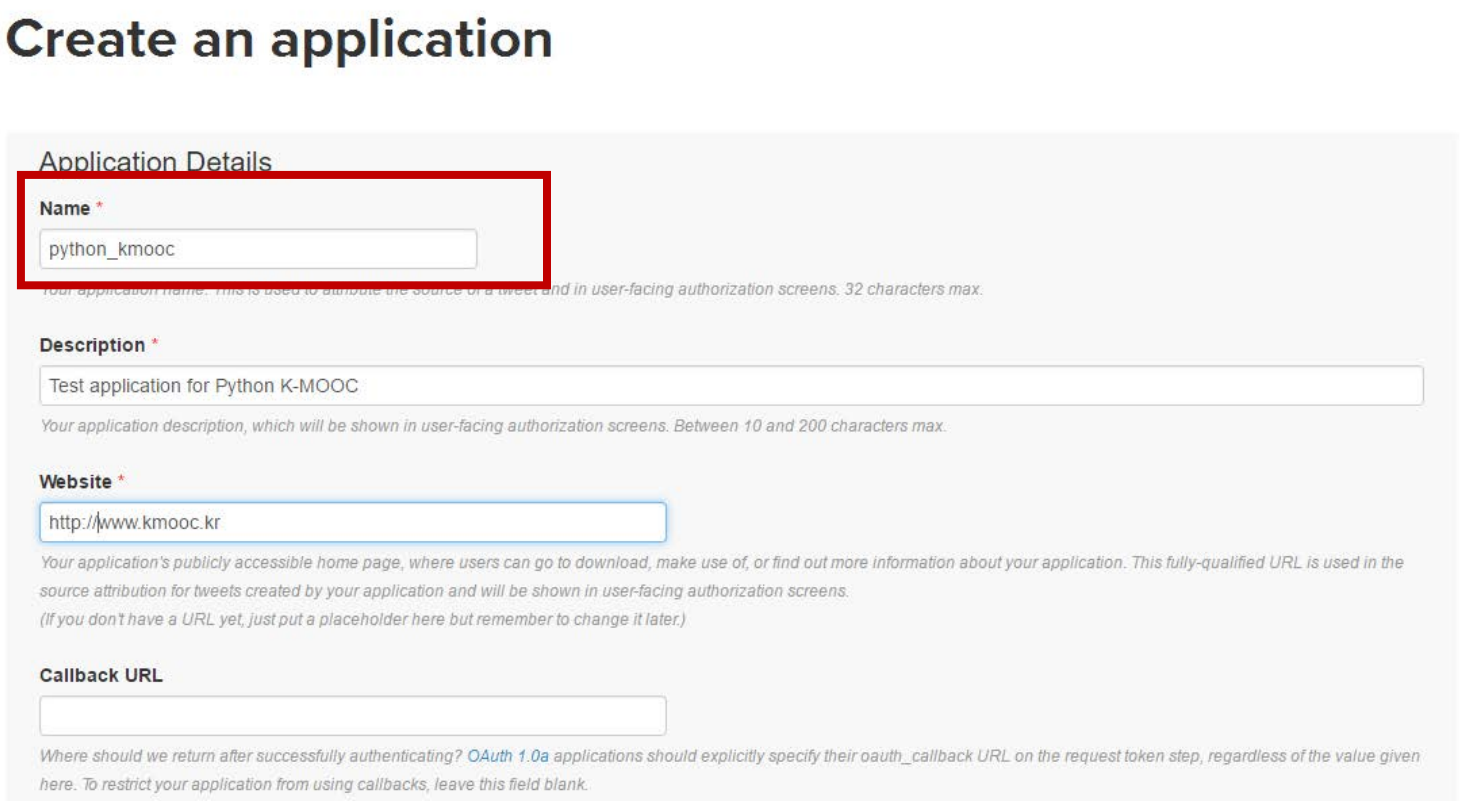
Twitter 데이터 가져오기 http://jinse.datastats.info/1

- 트위터 가입후 Twitter App 생성 https://apps.twitter.com/



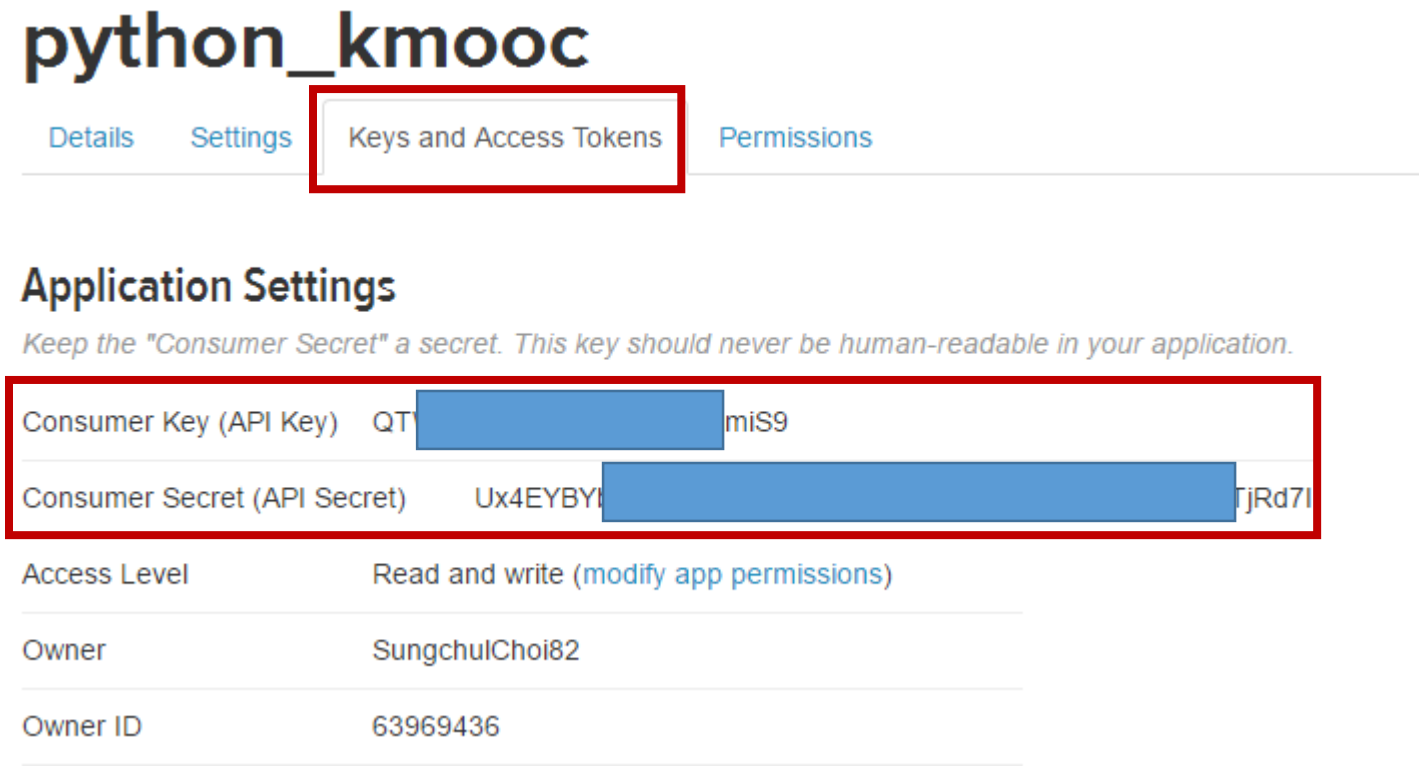
Twitter 데이터 가져오기 http://jinse.datastats.info/1

- 트위터 App 정보 입력



Twitter 데이터 가져오기 http://jinse.datastats.info/1

- Keys와 Access Tokens로 가서 API Key 값 확인



Twitter 데이터 가져오기 http://jinse.datastats.info/1

- Keys와 Access Tokens로 가서 API Key 값 확인

API 사용을 위한 모듈 설치

- conda 가상 환경으로 requests 와 oauthlib 설치

```
acitvate python_mooc
conda install requests
pip install requests-oauthlib
```

Code http://jinse.datastats.info/1

- oauth 접속 권한 받기

```
import requests
from requests_oauthlib import OAuth1

consumer_key = '확인한 consumer_key'
consumer_secret = '확인한c onsumer_secret'
access_token = '확인한 access_token'
access_token_secret = '확인한 access_token_secret'

oauth = OAuth1(client_key=consumer_key, client_secret=consumer_secret,
               resource_owner_key=access_token, resource_owner_secret=access_token_secret)
```

Code http://jinse.datastats.info/1

- 특정 계정의 타임라인 데이터 가져오기

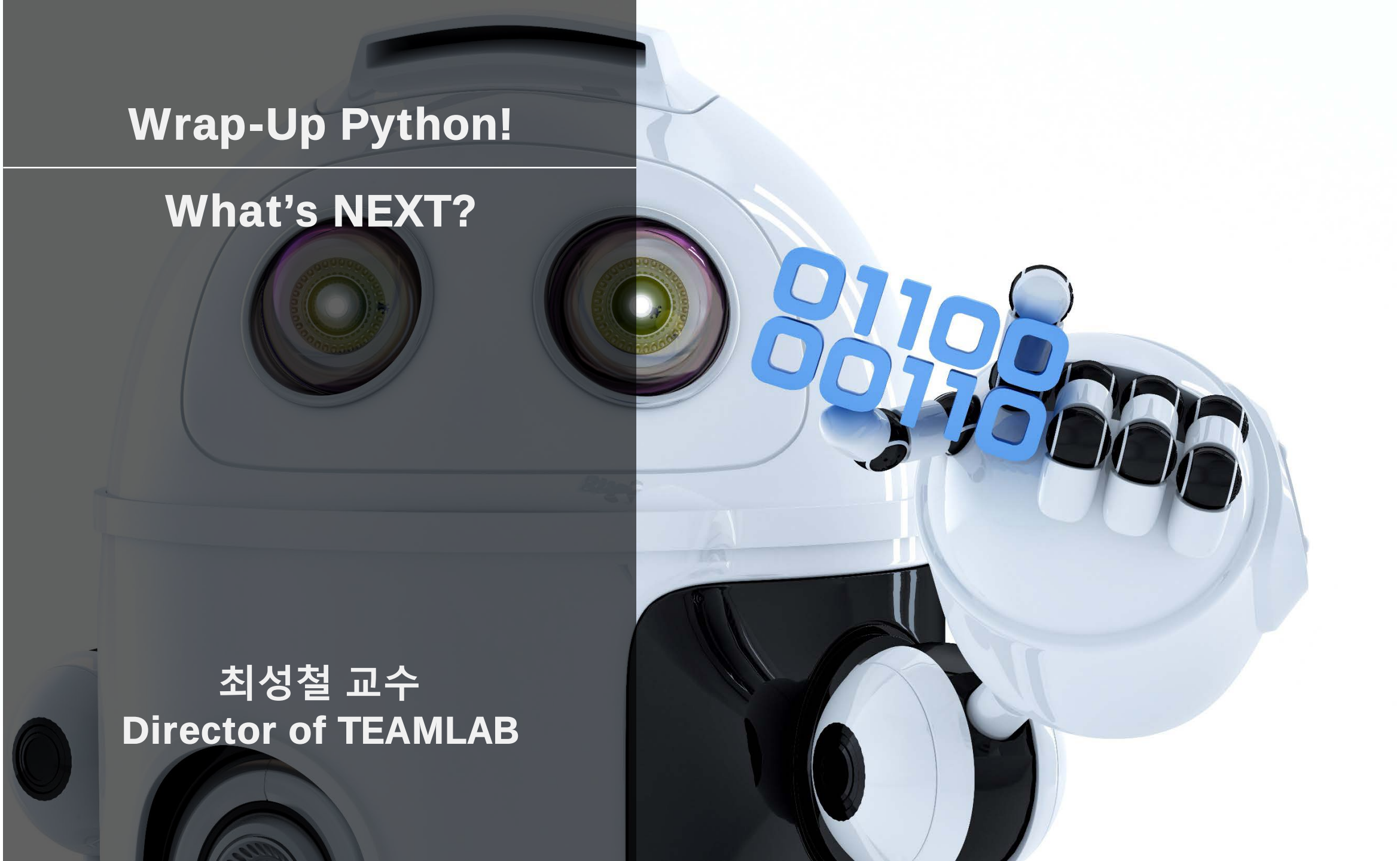
```
# Twitter REST api // screen_name 은 트위터 계정명
url = 'https://api.twitter.com/1.1/statuses/user_timeline.json?screen_name={0}'.format('naver_d2')
r = requests.get(url=url,auth=oauth)
statuses =  r.json()

for status in statuses:
    print (status['text'], status['created_at'])
```

Human knowledge belongs to the world.

Wrap-Up Python!

What's NEXT?



최성철 교수
Director of TEAMLAB

이제 기본은
다 배웠음

이제 뭘 해야 할까?

입문자 to 숙련자

파이썬 스터디의 주요 분야

- Data 분석: 머신러닝, 통계, Visualization
- 웹 프로그래밍: 웹 프레임워크
- 파이썬 성능 향상: 동시성, 프로파일링
- 코드 작성 & 협업: 파이썬 문서화, Github

Data 분석

Scikit-Learn – 머신러닝 라이브러리 http://scikit-learn.org/

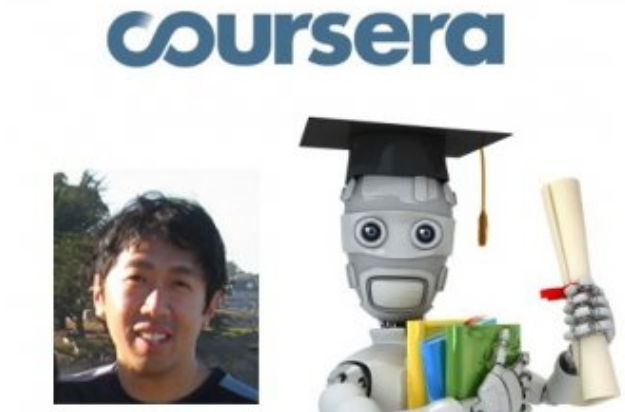
matplotlib – 데이터 시각화 라이브러리 http://matplotlib.org/

numpy – 과학 연산을 위한 라이브러리 http://www.numpy.org/

pandas – 데이터 Hadling을 위한 라이브러리 http://pandas.pydata.org/

Tensorflow – 딥러닝/머신러닝 라이브러리 https://www.tensorflow.org/

Data 분석



- 스탠포드 Andrew Ng 교수님 강의
- Coursera, 머신러닝 분야의 정석
- Matlab to Python 도전 권장

https://www.coursera.org/learn/machine-learning

- 밑바닥부터 시작하는 데이터 과학
- 파이썬의 데이터 분석 기초 과정
- 학부 수준 통계, 확률, 선형대수 이해 필수

Data 분석

- Coding the Matrix (Coursera)
- 브라운 대학의 Phil Klein
- 선형대수학을 Python으로 배우는 과정

http://codingthematrix.com/

- 홍콩 과기대 김성훈 교수님
- 모두를 위한 머신러닝/딥러닝 강의
- 한국어로 된 최고의 딥러닝 입문 과정

https://hunkim.github.io/ml/

Data 분석

https://www.inflearn.com/course/%ED%8C%8C%EC%9D%B4%EC%8D%AC-%EB%A8%B8%EC%8B%A0%EB%9F%AC%EB%8B%9D-%EC%9E%85%EB%AC%B8-%EA%B0%95%EC%A2%8C/

웹 프로그래밍

Django – 가장 넓게 쓰이는 파이썬 웹 프레임워크
https://www.djangoproject.com/

flask – 경량 파이썬 웹 프레임워크, Easy & Simple
http://flask.pocoo.org/

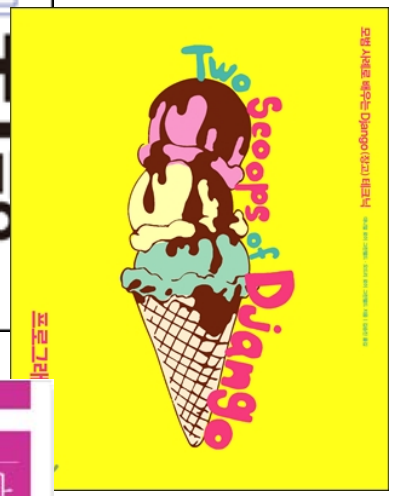
웹 프로그래밍

- django Girls Tutorial
- https://tutorial.djangogirls.org/ko/
- 누구나 쉽게 웹 프로그래밍 시작하기

- ask Django
- https://www.facebook.com/groups/askdjango/
- Django의 다양한 개발 사례를 공유
- 오프라인 세미나 개최

웹 프로그래밍

- 파이썬 웹 프로그래밍 실전편
- 기초부터 실전까지 입문서로 추천



- Two Scoops of Django
- Django 중급 이상을 위한 좋은 가이드북

- 클린 코드를 위한 테스트 주도 개발
- Django를 복습하면서 UnitTest 이해를 위한 책

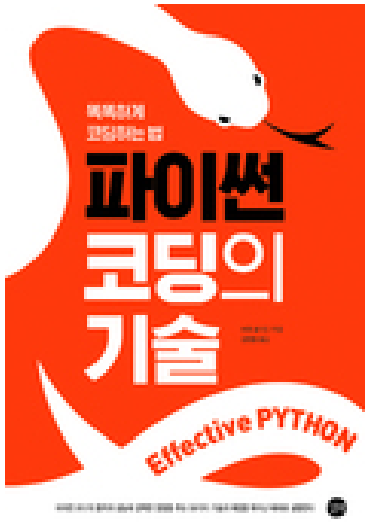
파이썬 성능향상

동시성 – 한번에 한 가지 이상의 Task를 실행시키기!

profile – 메모리/연산이 많은 지점 확인 하기

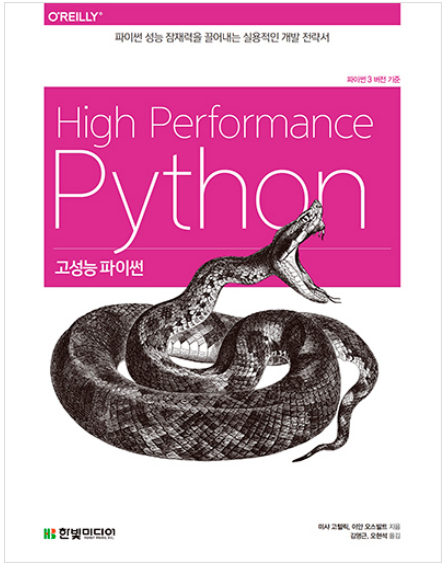
파이썬 성능향상

Fluent Python – 전문가를 위한 파이썬
(한빛미디어, 2016)



Effective Python - 파이썬 코딩의 기술
(길벗, 2016)

파이썬 성능향상



High Performance Python - 고성능 파이썬
(한빛미디어, 2016)

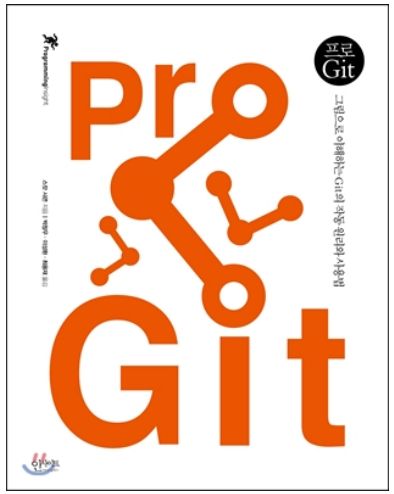
깐깐하게 배우는 파이썬
(인사이트, 2016)

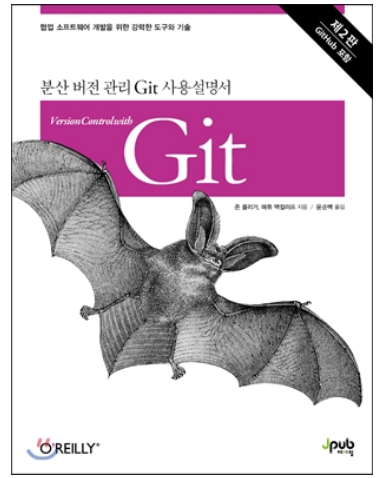
파이썬 성능향상

파이썬을 여행하는 히치하이커를 위한 안내서
(인사이트, 2017)

프로그래밍 잘하기

Git, Github study – 같이 개발하는 방법 배우기

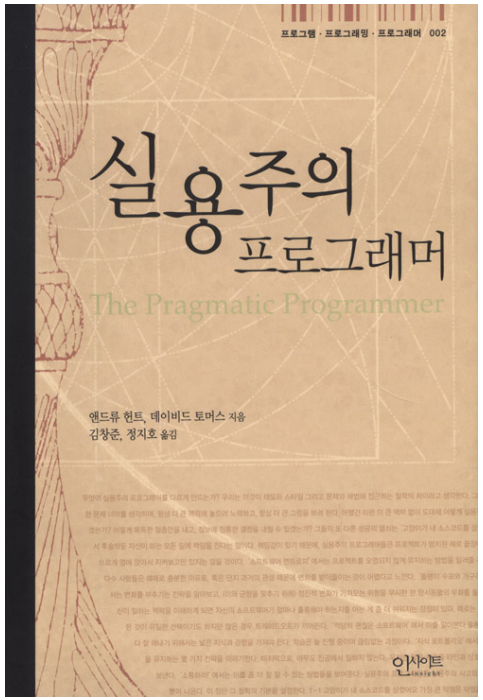


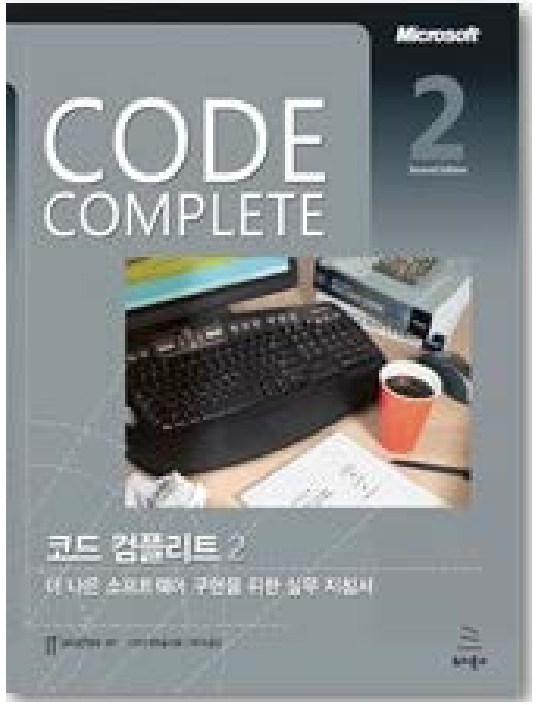


- 가천대학교 CS50 강좌 - https://goo.gl/FTrK90
- 홍콩과기대 김성훈 교수 github flow
  https://goo.gl/t9K8gn, https://goo.gl/Ek35Zi

프로그래밍 잘하기

프로그래밍 자체에 대한 공부





Human knowledge belongs to the world.