

오픈API 활용한 데이터 수집

김효관 |

웹 크롤링 이해

웹 크롤링의 이해

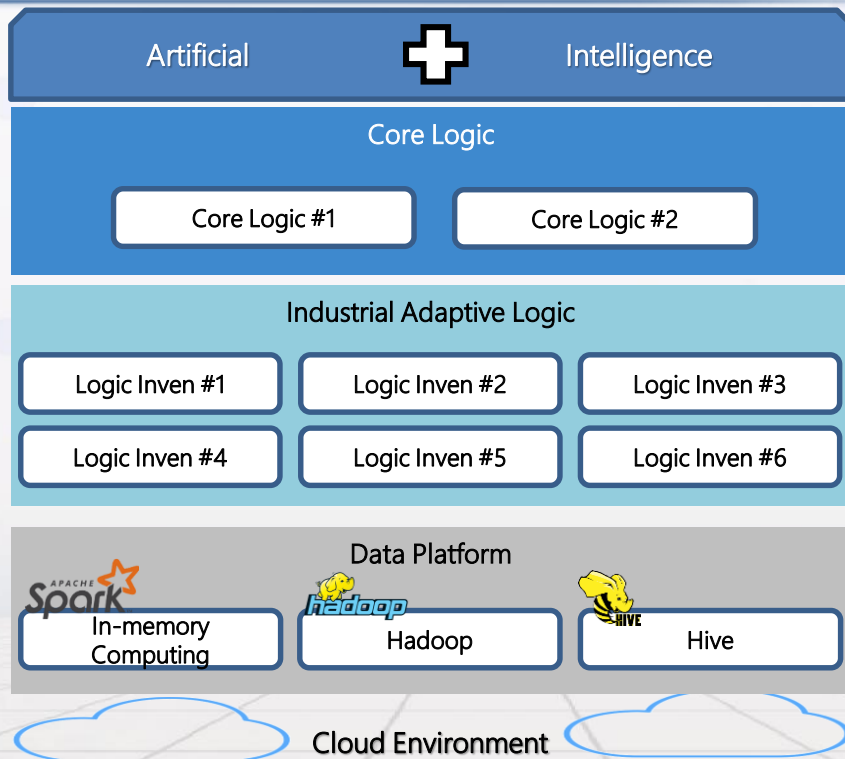
우리 회사에 필요한 가치를 창출하기 위해 구성도를 그린다...

분석 데이터

내부 데이터



이미 사내에서
접속 가능한
정제 후 활용 가능 데이터



시각화 / 웹 시연



웹 크롤링의 이해

시간이 지나자... 조금 한계에 ...



가지고 있는 걸로 분석 해보자..

1일...

2일...

3일...

하아... 가지고 있는 데이터만으로는 다양한 각도로 분석을 할 수 없네.....

웹 크롤링의 이해

외부 데이터에 필요한 정보들을 수집해보자!

분석 데이터

내부 데이터



외부 데이터



Analytics



Intelligence

Core Logic

Core Logic #1

Core Logic #2

Industrial Adaptive Logic

Logic Inven #1

Logic Inven #2

Logic Inven #3

Logic Inven #4

Logic Inven #5

Logic Inven #6

Data Platform



In-memory
Computing



Hadoop



Hive

Cloud Environment

시각화 / 웹 시연



웹 크롤링의 이해

데이터가 많아지니 분석해볼 만한게 많아지네!

웹 상에 공개된 데이터 스크랩 후 활용

접속 사이트	제공항목(데이터셋)
국가통계포털	코스닥지수
국가통계포털	코스닥 150 지수
국가통계포털	코스닥 주가이익비율 (PER)
국가통계포털	코스닥 주가순자산비율 (PBR)
국가통계포털	코스닥 배당수익률
국가통계포털	코스닥 산업별 투자지표
공공데이터포털	한국감정원 오피스텔 동향조사 현황
공공데이터포털	한국감정원 부동산 매매가격지수 현황
공공데이터포털	공간융합정보

내부 데이터 정제 후 활용



데이터 탐색 후 필요 데이터 직접 스크랩 후 활용

timeanddate.com

Home / World Clock / Time Zones / Calendar / Weather / Sun & Moon / Timers / Calculators

Home / Calendar / Holidays / South Korea

🇰🇷 Holidays and Observances in South Korea in 2019

Suggested countries: Australia | Canada | United Kingdom | United States | List of all countries

Showing (30): All holidays/observances For: 2019

Jump to: [Next](#) | [JAN](#) | [FEB](#) | [MAR](#) | [APR](#) | [MAY](#) | [JUN](#) | [JUL](#) | [AUG](#) | [SEP](#) | [OCT](#) | [NOV](#) | [DEC](#)

Holidays and Observances in South Korea in 2019


Date	Name	Type
1 Jan	Tuesday New Year's Day	Public Holiday
4 Feb	Monday Seollal Holiday	Public Holiday



파이썬 개발환경 설치!



앞으로 웹에서 데이터를 가져오는 방법을 배웁니다.
그전에! **핵심문법**을 둘러보겠습니다.
(**이미 배웠다면 Pass!!!**)



Python 기본문법

김효관 |



교육목표: Python 기본 문법 조작 방법을 익힌다.

CONTENTS

1

Python 기본문법



1. 파이썬 기본문법

주요 문법

1-1 반복하기

1-2 조건 판단하기

1. 파이썬 기본문법

1. 반복하기 (while, for)



```
tvList = [ UN40EN001, UN40EN002, UN40EN003, UN40EN004]
```

헉! tv목록앞에 제품목록을 전부 붙여야하는데.... 어떻게하지?
하나씩 하면 되겠네..

```
preFix = "LEDTV_"  
tvList[0] = prefix + tvList[0]  
tvList[1] = prefix + tvList[1]
```

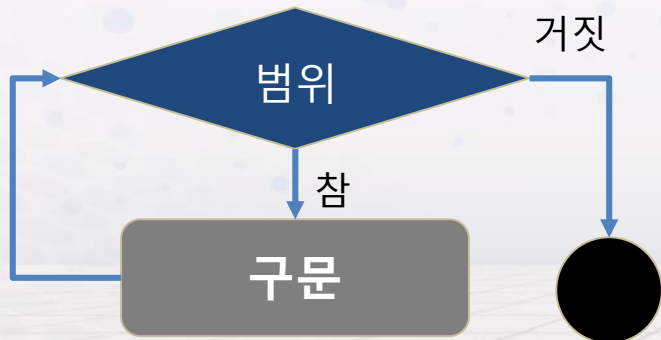
만약.. 목록이 십만개가 넘는다면??

1. 파이썬 기본문법

1. 반복하기 (for, while)

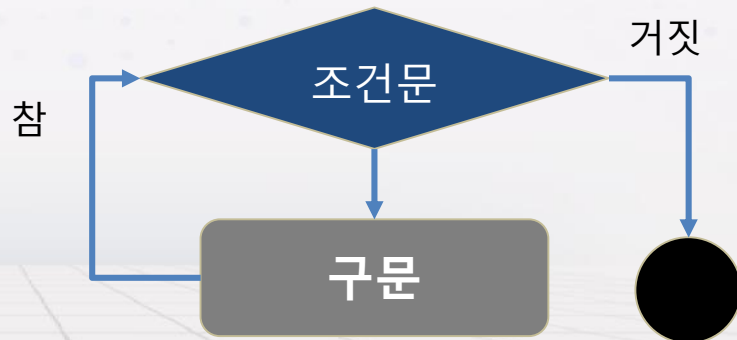
주어진 범위 안에서
구문 수행

for



조건문이 참인 경우 조건문 이하 구문을 수행
만약 거짓이면 구문 수행하지 않음

while



1. 파이썬 기본문법

1. 반복하기 (while, for)

while (조건) loop : 조건 실패 시 조건 문 탈출
for (조건) : 조건 실패 시 조건 문 탈출

Example

```
tvList = [ 'UN40EN001', 'UN40EN002', 'UN40EN003', 'UN40EN004' ]
```

for 문

```
preFix = "LEDTV_"  
### range: last index is not included  
### list(range(0,4))  
for i in range(0,4):  
    tvList[i] = preFix + tvList[i]
```

for문은 변수가 특정 범위안에서
구문연산 (구문은 탭으로 한칸띄)

while 문

```
preFix = "LEDTV_"  
### range: last index is not included  
### list(range(0,4))  
i=0  
listLength = len(tvList)  
while(i < listLength ):  
    tvList[i] = preFix + tvList[i]  
    i = i+1
```

while문은 조건식을
만족하는 동안 구문연산

```
tvList2 = [ 'UN40EN001', 'UN40EN002',  
            'UN40EN003', 'UN40EN004']  
리스트가 출력되도록 하세요
```

힌트: `print(tvList2[0])`

1. 파이썬 기본문법

2. 조건 판단하기



```
tvList = [ UN40EN001, LEDTV_UN40EN002,  
            LEDTV_UN40EN003, UN40EN004]
```

헉! tv목록앞에 제품목록을 전부 붙여야하는데.... 붙어있는게 있고 없는게 있네.. 어떻게 하지?
붙어 있는지 조건을 판단해야겠는데...

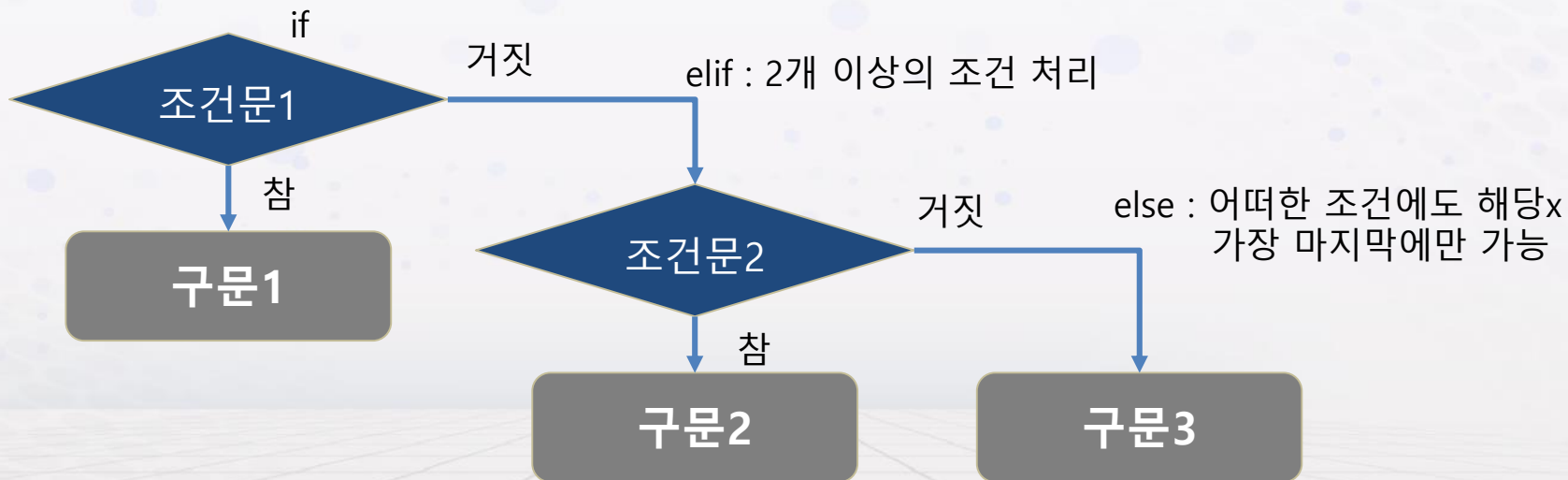
(만약, LEDTV_가 있으면 넘기고... LEDTV_가 없으면 넣고..)

조건문은 어떻게 사용할 수 있을까요?

1. 파이썬 기본문법

2. 조건 판단하기

조건문을 평가하고 참인 경우만 구문이 수행
2개 이상의 구문은 들여쓰기로 블록을 지정



1. 파이썬 기본문법

2. 조건 판단하기

```
if [조건문]: # 조건이 일치하는 경우 실행
    [조건문 참인경우 실행]
else:
    [조건문 불일치 시 실행]
```

Example

```
tvList2 = [ 'UN40EN001', 'LEDTV_UN40EN002', 'LEDTV_UN40EN003', 'UN40EN004']
```

```
preFix = "LEDTV_"
### range: last index is not included
### list(range(0,4))
for i in range(0,4):
    preFixCount = tvList2[i].count(preFix)
    if(preFixCount == 0):
        tvList2[i] = preFix + tvList2[i]
    else:
        tvList2[i] = tvList2[i]
```

미션: 모든 목록에 LEDTV를 붙여야한다. (단, 이미 붙어있으면 제외한다)

```
tvList2 = [ 'UN40EN001', 'LEDTV_UN40EN002', 'LEDTV_UN40EN003', 'UN40EN004']
```

```
preFix = "LEDTV_"
### range: last index is not included
### list(range(0,4))
for i in range(0,4):
    preFixCount = tvList2[i].count(preFix)
    if(preFixCount == 0):
        tvList2[i] = preFix + tvList2[i]
    else:
        tvList2[i] = tvList2[i]
```

```
tvList2
```

교육목표: Python을 활용하여 공개된 데이터를 불러올 수 있다.

CONTENTS

1 공공데이터 포털 파일 데이터 불러오기

2 통계청 포털 API 데이터 불러오기

3 공공데이터 포털 API 데이터 불러오기

4 핵심정리 및 Q&A



1. 공공데이터 포털 파일 데이터 불러오기

1. 공공데이터 포털 파일 데이터 불러오기

데이터타입 데이터 불러오는 순서

- 1 접속 및 회원가입
- 2 자료검색 및 제공유형 확인
- 3 데이터 다운로드 URL 확인
- 4 파이썬 활용 데이터 불러오기

1. 공공데이터 포털 파일 데이터 불러오기

1. 접속: (<https://www.data.go.kr/>) 및 회원가입

The screenshot shows the homepage of the Korea Public Data Portal. At the top, there is a navigation bar with the 'DATA GO.KR' logo, a '로그인' (Login) button, a '회원가입' (Sign Up) button, a '사이트맵' (Site Map) link, an 'ENGLISH' toggle, and social media icons for Facebook, Twitter, and YouTube. Below this is a secondary navigation bar with links for '데이터셋' (Data Sets), '제공신청' (Request for Provision), '활용사례' (Use Cases), '정보공유' (Information Sharing), and '이용안내' (Usage Guide). A search bar is prominently displayed with the text '데이터를 검색해보세요!' (Search for data!) and a search button. The search results show '주택금융관련 지수' (Real Estate Finance Related Index) as the top result. Below the search bar, there is a section for '최근 사회현안 및 이슈' (Recent Social Issues and Issues) with a '미세먼지' (PM2.5) icon and the text '데이터를 확인해 보세요!' (Check the data!). The main content area is divided into two sections: '국가중점데이터' (National Key Data) and '데이터 카테고리' (Data Category). The '국가중점데이터' section lists various data categories with icons: 건축정보 (Construction Information), 교통사고정보 (Traffic Accident Information), 국민건강정보 (National Health Information), 상권정보 (Commercial Area Information), 수산정보 (Fisheries Information), 실시간수도정보 (Real-time Water Information), 농수축산가격정보 (Agriculture, Forestry, and Fisheries Prices Information), and 등산로 정보 (Hiking Trail Information). The '데이터 카테고리' section lists more categories: 부동산종합정보 (Real Estate Comprehensive Information), 통합재정정보 (Integrated Financial Information), 지방행정정보 (Local Government Information), 부동산거래정보 (Real Estate Transaction Information), 식의약품종합정보 (Food and Drug Comprehensive Information), 지방재정정보 (Local Finance Information), 법령정보 (Regulatory Information), and a '+ 더보기' (More) button. At the bottom, there are three promotional banners. The first banner on the left is titled '데이터1번가' (Data 1st Street) and promotes the ease of accessing public data. The middle banner is titled '이슈 데이터 미세먼지 정보' (Issue Data PM2.5 Information) and features a map of Korea. The right banner is titled '대한민국 공공데이터 개방' (Korea Public Data Opening) and celebrates the '2회연속 세계 1위 달성!' (Achieving 1st place in the world for 2 consecutive years!).

DATA 공공데이터포털 .GO .KR

로그인 회원가입 사이트맵 ENGLISH f t B

데이터셋 제공신청 활용사례 정보공유 이용안내

데이터를 검색해보세요! 주택금융관련 지수 🔍

2.지하철 ▼

최근 사회현안 및 이슈 미세먼지 데이터를 확인해 보세요!

국가중점데이터 데이터 카테고리

건축정보 교통사고정보 국민건강정보 상권정보 수산정보 실시간수도정보 농수축산가격정보 등산로 정보

부동산종합정보 통합재정정보 지방행정정보 부동산거래정보 식의약품종합정보 지방재정정보 법령정보 + 더보기

「데이터1번가」
여러분이 필요로 하는 공공데이터를 자유롭게
신청할 수 있는 데이터 소통창구

이슈 데이터 미세먼지 정보
미세먼지 데이터
미세먼지와 관련된 공공데이터 결합정보를
쉽게 확인할 수 있습니다.

대한민국 공공데이터 개방
2회연속 세계 1위 달성!

1. 공공데이터 포털 파일 데이터 불러오기

2. 자료검색 및 제공유형 확인

검색어로 데이터 탐색

DATA 공공데이터포털 .GO .KR 로그인 회원가입 사이트맵

데이터셋 제공신청 활용사례 정보공유

주택금융관련 지수

1. 자료 검색

전체(1) 파일데이터(1) 오픈API(0) 표준데이터(0)

전체 1건을 찾았습니다.

파일데이터 [1건] 정확도 ▾ 날짜 제목 조회수 다운로드

주택금융관련 지수 조회수: 702 다운로드수: 1,596
수정일: 2018.01.03 기관: 한국주택금융공사 서비스유형: 다운로드
주택금융관련 지수 CSV

주택금융관련 지수 조회수: 702 다운로드수: 1,596
수정일: 2018.01.03 기관: 한국주택금융공사 서비스유형: 다운로드
주택금융관련 지수 CSV

전체 1건을 찾았습니다.

파일데이터 [1건] 정확도 ▾ 날짜 제목 조회수 다운로드

주택금융관련 지수 조회수: 702 다운로드수: 1,596
수정일: 2018.01.03 기관: 한국주택금융공사 서비스유형: 다운로드
주택금융관련 지수 CSV

재정금융

2. 파일 제공여부 확인 (1건 제공)

1. 공공데이터 포털 파일 데이터 불러오기

3. 데이터 다운로드 url 확인

다운로드 URL 확인

주택금융관련 지수

매체유형: 텍스트 파일, 링크 건수: 22 전체 행수: N/A 확장자: CSV 다운로드 횟수(바로그기 횟수): 1596

☐ 전체

선택 다운로드

※ 서비스 오류가 있을시 오류신고 버튼을 이용해주세요.

☐

CSV

(2017)지역별 연간소득대비 주택가격...

☐

CSV

(2017)주택규모별 주택구입부담지수

멀티다운로드

Q 찾기

오류신고

★

멀티다운로드

상세정보

오류신고

★

(2017)지역별 연간소득대비 주택가격 비율 및 연간소득 대비 대출금액비율

업데이트 주기	수시	차기등록예정일	2018-12-28
비용부과유무	무료	비용부과기준 및 단위	없음
다운로드 횟수	373		
등록일	2017-12-28	수정일	2018-01-03
이용허락범위	이용허락범위 제한 없음		
제공형태	공공데이터포털에서 다운로드(원문파일등)		
URL	https://www.data.go.kr/dataset/fileDownload.do?atchFileId=FILE_000000001420537&fileDetailSn=1		
설명	국내 지역별 PIR 및 LIR 정보 제공 (PIR(Price Income Ratio) = 평균 주택가격 / 평균 연소득), (LIR(Loan Income Ratio) = 평균 대출금액 / 평균 연소득)		
관리부서명	ICT전략부	관리부서 전화번호	

속성



fileDownload.do?

atchFileId=FILE_000000001420537&fileDetailSn=1&

프로토콜: HyperText Transfer Protocol with Privacy

유형: 다운로드용

주소:

https://www.data.go.kr/dataset/fileDownload.do?
(URL) atchFileId=FILE_000000001420537&fileDetailSn=1&

1. 다운로드 url

확인

취소

적용(A)

- 다운로드 url 전체 url을 확인을 위해 우클릭 속성 → url 확인필요

1. 공공데이터 포털 파일 데이터 불러오기

4. 파일썸 활용 데이터 불러오기

```
pd.read_csv("url")
```

3. 파일데이터 불러오기

```
import pandas as pd
```

```
url = 'https://www.data.go.kr/dataset/fileDownload.do?atchFileId=FILE_000000001420537&fileDetailSn=1&publicDataO'
```

```
data = pd.read_csv(url, encoding='ms949')
```

```
data.head()
```

	연도	월	전국 PIR	서울 PIR	부산 PIR	대구 PIR	인천 PIR	광주 PIR	대전 PIR	울산 PIR	...	세종 LIR	경기 LIR	강원 LIR	충북 LIR	충남 LIR	전북 LIR	전남 LIR	경북 LIR	경남 LIR	제주 LIR
0	2004	3	4.21	4.89	3.95	3.73	4.65	2.81	4.68	2.66	...	-	2.42	1.44	1.53	1.21	1.90	1.42	1.31	2.06	1.28

공공데이터포털에서 한국감정원 오피스텔 동향조사 현황 자료를 검색하여 데이터타입 자료를 불러오세요

한국감정원 오피스텔 동향조사 현황

한국감정원에서 조사하는 오피스텔 동향조사 중 오피스텔 매매, 전세, 월세가격

매체유형 : 텍스트 파일, 링크 건수 : 16 전체 행 수 : N/A 확장자 : CSV

☐ 전체

선택 다운로드

☒

CSV

2018년 1월~2018년 2월 오피스텔 ...

멀티다운로드

닫기

오류신고

	지역	2018년 01월 평균매매가격	2018년 01월 평균단위당매매가격	2018년 01월 중위매매가격	2018년 01월 중위단위당매매가격
0	전국	176088	4293	155641	4185
1	수도권	189101	4710	170113	4595
2	지방	121651	2546	95096	2471
3	서울	221357	5809	198135	5724
4	서울 도심권	281658	6783	265382	7090
5	서울 동북권	150520	5026	151166	4633
6	서울 서북권	200156	6085	188176	6051
7	서울 서남권	186320	4961	166027	4762

2. 통계청 포털 API 데이터 불러오기

2. 통계청 포털 API 데이터 불러오기 (JSON)

소개

- 1 공식 사이트 접속
- 2 OpenAPI 사이트 접속 및 회원가입
- 3 통계자료 활용신청
- 4 통계표 URL 생성
- 5 Json 데이터 이해
- 6 Python 활용 불러오기

2. 통계청 포털 API 데이터 불러오기 (JSON)

1. 사이트 접속: (http://kosis.kr/index/index.do)



행정구역별 성별 인구수 조회

접속 사이트 제공항목(데이터셋)

국가통계포털 행정구역(시군구)별, 성별 인구수

폴더명

0_행정구역(시군구)별, 성별 인구수

2. OpenApi 사이트 접속 및 회원가입

<https://kosis.kr/openapi/index/index.jsp>



1. OpenAPI 접속

<http://kosis.kr/openapi>

KOSIS 공유서비스에서는 KOSIS에 수록된 통계정보를 이용하여 공공 및 민간 등에서 자체적으로 서비스를 개발할 수 있도록 국가통계통합DB에 접근하기 위한 인터페이스(API)를 제공합니다.

KOSIS 통계정보(통계목록, 통계자료, 내용량통계자료, 통계설명, 통계표설명, KOSIS통합검색, 통계주요지표)를 JSON, SDMX, XML, XLS와 같이 다양한 형태로 이용하실 수 있으며, 통계지표 OpenAPI 서비스 또한 이용하실 수 있습니다.

자는 통계 기반 웹사이트 구축, DB 연동, 모바일 앱 개발 등에 활용할 수 있습니다.



2. 통계청 포털 API 데이터 불러오기 (JSON)

3. 통계자료 활용신청

<https://kosis.kr/openapi/index/index.jsp>

1. 메뉴이동

공유서비스 소개 및 이용방법 서비스이용 개발 가이드 커뮤니티 이용현황

통계목록 통계자료 대용량통계자료 통계설명 통계표설명 KOSIS통합검색 통계주요지표

통계자료 통계자료를 제공합니다. [제공정보 확인하기](#) Home > 서비스 이용 > 통계자료 인쇄

활용신청 신청현황 자료등록 URL생성

자료등록 등록된자료

작성기관 통계표명 분류 분류값/항목 통계조사명 통계표ID 주기 등록한 표만 보기

인구수 DT_1B040A3 전체 검색

기관명	통계표명	통계표ID	통계조사명	수록기간	사용여부	통계표조회
통계청 (101)	행정구역(시군구)별, 성별 인구수	DT_1B040A3	주민등록인구현황	월,년 1992~2019.06	<input type="checkbox"/>	

통계표등록

2. 활용신청

신청시 목적은
일반적인
내용으로 작성!

2. 통계청 포털 API 데이터 불러오기 (JSON)

4. 통계표 URL 생성

<https://kosis.kr/openapi/index/index.jsp>

자료등록

활용신청 > 신청현황 > 자료등록 > URL생성

자료등록 | 등록된자료

작성기관: 통계표명: 통계표ID: 분류: 주기: 검색

1. 자료 조회

기관명	통계표명	통계표ID	통계조사명	수목기간	사용여부	통계표조회
통계청 (101)	인구수별 행정리 수	DT_1AR15108	농림어업총조사	5년 2015~2015	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	행정구역(시군구)별, 성별 인구수	DT_1B040A3	주민등록인구현황	월, 년 1992~2019.06	<input checked="" type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	시도/가정원수/경제활동 인구수별 가구	DT_1EC7501	인구총조사	5년 1975~1975	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	시도/가정원수/경제활동 인구수별 가구	DT_1EC800F	인구총조사	5년 1980~1980	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	계합수면 세대 및 인구수	DT_1IN2505	인구총조사	5년 1925~1925	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	행정구역(시/국/국립) 세대 및 인구수	DT_1IN4407	인구총조사	5년 1944~1944	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	인구수	DT_2AS033	World Bank	년 1960~2017	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	가정원 인구수	DT_1ZGA02_012	1993년 북한인구 일제 조사	부정기 1993~1993	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	15-49세 여성인구수 및 비율	DT_1ZGI01_106	북한통계(UN:국제연합) 국제기통계	년 1950~2100	<input type="checkbox"/>	<input type="checkbox"/>
통계청 (101)	나이별, 성별 인구수, 남녀별 비중, 설비	DT_1ZGI05_001	2008년 북한인구 일제 조사	부정기 2008~2008	<input type="checkbox"/>	<input type="checkbox"/>

2. 통계표 등록

URL 생성

통계자료 통계자료를 제공합니다. 제공정보 확인하기

Home > 서비스 이용 > 통계자료 > 안내

활용신청 > 신청현황 > 자료등록 > URL생성

※ 자료등록이 완료된 통계표에 대해서만 URL 생성 작업이 가능합니다.

작성기관: 통계조사명: 통계표ID: 분류: 주기: 검색

기관명	통계표명	통계표ID	통계조사명	수목기간	신청일자	URL 생성
통계청 (101)	행정구역(시군구)별, 성별 인구수	DT_1B040A3	주민등록인구현황	월, 년 1992~2019.06	2019-07-10 09:47:03	<input type="button" value="URL생성"/>

3. URL 생성

2. 통계청 포털 API 데이터 불러오기 (JSON)

4. 통계표 URL 생성

<https://kosis.kr/openapi/index/index.jsp>

통계자료 통계자료를 제공합니다. [제공정보 확인하기](#)

Home > 서비스 이용 > 통계자료 [인쇄](#)

[활용신청](#) [신청현황](#) [자료등록](#) [URL생성](#)

URL생성

URL생성 조건 설정

통계표명 행정구역(시군구)별, 성별 인구수

활용 자료명 **시군구별 인구수**

조회구분 [?](#) ☐ 시계열(time series) ☒ 횡단면(cross sectional)

분류

행정구역(시군구)별 ☒ 개별 ☐ 레벨별 1 / 2

항목

☒ 총인구수
☐ 남자인구수
☐ 여자인구수

분류/항목선택

[URL생성](#) 취소

1. 활용 자료명 입력

2. 분류 및 항목 선택

3. URL 생성

통계자료 개발가이드 | KOSIS공유서비스 - Chrome

https://kosis.kr/openapi/abroad/treelist/detail_vw_setjsp?org_id=101...

분류값 선택

행정구역(시군구)별 <= 1 / 2 [이동](#)

코드 분류값 [검색](#)

<input type="checkbox"/>	코드	분류값
<input checked="" type="checkbox"/>	00	전국
<input checked="" type="checkbox"/>	11	서울특별시
<input checked="" type="checkbox"/>	26	부산광역시
<input checked="" type="checkbox"/>	27	대구광역시
<input checked="" type="checkbox"/>	28	인천광역시
<input checked="" type="checkbox"/>	29	광주광역시
<input checked="" type="checkbox"/>	30	대전광역시
<input checked="" type="checkbox"/>	31	울산광역시
<input checked="" type="checkbox"/>	36	세종특별자치시
<input checked="" type="checkbox"/>	41	경기도
<input checked="" type="checkbox"/>	42	강원도

[선택](#) [닫기](#)

2. 통계청 포털 API 데이터 불러오기 (JSON)

4. 통계표 URL 생성

활용신청

신청현황

자료등록

URL생성

URL생성

URL생성 조건 설정

통계표명	행정구역(시군구)별, 성별 인구수	통계표ID	DT_1B040A3
조회구분	횡단면		

URL생성 상세조건 정보

출력형태 설정	
데이터 포맷	<input checked="" type="radio"/> JSON <input type="radio"/> SDMX

조회기간 설정 ☐ 기간설정 ☒ 최근시점기준

최근시점부터	주기 월	최근 1개 자료
수룩시점간격		최근시점 조정에 사용할 수 있는 항목으로 현재 시점부터 사용할 기준시점까지의 간격을 입력 ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력

활용자료명	URL 상세 정보	조회구분	URL 보기	결과값보기
인구수	총인구수 * 강원도 / 경기도 / 경상남도 / 경상북도	횡단면	URL보기	결과값보기

생성된 URL

http://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=MjUzMtDiMjcyZjQwY2VkMWJkY2RkYTRmZDkwMWQ3ZGY=&format=json&jsonVD=Y&userStatsId=haiteam/101/DT_1B040A3/2/1/20190710100719&prdSe=M&newEstPrdCnt=1

URL 복사

목록

URL 보기 선택 후
URL 복사

http://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=MjUzMtDiMjcyZjQwY2VkMWJkY2RkYTRmZDkwMWQ3ZGY=&format=json&jsonVD=Y&userStatsId=haiteam/101/DT_1B040A3/2/1/20190710100719&prdSe=M&newEstPrdCnt=1

2. 통계청 포털 API 데이터 불러오기 (JSON)

5. JSON 데이터 이해

<https://kosis.kr/openapi/index/index.jsp>

```
[{"TBL_NM": "업종 < 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9 / 10 / 11 / 12 / 13 / 14 / 15 / 16 / 17 / 18 / 19 / 20 / 21 / 22 / 23 / 24 / 25 / 26 / 27 / 28 / 29 / 30 / 31 / 32 / 33 / 34 / 35 / 36 / 37 / 38 / 39 / 40 / 41 / 42 / 43 / 44 / 45 / 46 / 47 / 48 / 49 / 50 / 51 / 52 / 53 / 54 / 55 / 56 / 57 / 58 / 59 / 60 / 61 / 62 / 63 / 64 / 65 / 66 / 67 / 68 / 69 / 70 / 71 / 72 / 73 / 74 / 75 / 76 / 77 / 78 / 79 / 80 / 81 / 82 / 83 / 84 / 85 / 86 / 87 / 88 / 89 / 90 / 91 / 92 / 93 / 94 / 95 / 96 / 97 / 98 / 99 / 100", "PRD_DE": "201812", "TBL_ID": "DT_1G1B002"}
```

• object 형식

객체. key/value 형태의 비순서화된 SET 구조이며 key는 string 타입으로 작성한다.



```
[ {TBL_NM": "ABC", "PRD_DE" : "201812"},  
  {TBL_NM": "ABC", "PRD_DE" : "201813"},  
  {TBL_NM": "ABC", "PRD_DE" : "201814"} ]
```

1행 데이터

2행 데이터

3행 데이터

2. 통계청 포털 API 데이터 불러오기 (JSON)

5. JSON 데이터 이해

Open API | KOSIS 국가통계포털 x | KOSIS 공유서비스 x | kosis.kr/openapi/statisticsData.do x +

← → ↺ ⓘ 주의 요함 | kosis.kr/openapi/statisticsData.do?method=getList&apiKey=MjUzMTdiMjc5ZjQwY2VkMWJkY2... ☆ 효관 ⋮

```
[{"TBL_NM": "월별총생산액(제조업)의 지역별 분포", "PRD_DE": "201906", "TBL_ID": "DT_1B040A3", "ITM_NM": "제조업", "ITM_NM_ENG": "Koreans (Total)", "ITM_ID": "T20", "UNIT_NM": "천원", "ORG_ID": "101", "UNIT_NM_ENG": "Person", "C1_OBJ_NM": "월별총생산액(제조업)의 지역별 분포", "C1_OBJ_NM_ENG": "By Administrative District", "DT": "9757144", "PRD_SE": "M", "C1": "11", "C1_NM": "서울특별시", "C1_NM_ENG": "Seoul"}, {"TBL_NM": "월별총생산액(제조업)의 지역별 분포", "PRD_DE": "201906", "TBL_ID": "DT_1B040A3", "ITM_NM": "제조업", "ITM_NM_ENG": "Koreans (Total)", "ITM_ID": "T20", "UNIT_NM": "천원", "ORG_ID": "101", "UNIT_NM_ENG": "Person", "C1_OBJ_NM": "월별총생산액(제조업)의 지역별 분포", "C1_OBJ_NM_ENG": "By Administrative District", "DT": "3427632", "PRD_SE": "M", "C1": "26", "C1_NM": "부산광역시", "C1_NM_ENG": "Busan"}, {"TBL_NM": "월별총생산액(제조업)의 지역별 분포", "PRD_DE": "201906", "TBL_ID": "DT_1B040A3", "ITM_NM": "제조업", "ITM_NM_ENG": "Koreans (Total)", "ITM_ID": "T20", "UNIT_NM": "천원", "ORG_ID": "101", "UNIT_NM_ENG": "Person", "C1_OBJ_NM": "월별총생산액(제조업)의 지역별 분포", "C1_OBJ_NM_ENG": "By Administrative District", "DT": "2450478", "PRD_SE": "M", "C1": "27", "C1_NM": "대구광역시", "C1_NM_ENG": "Daegu"}, {"TBL_NM": "월별총생산액(제조업)의 지역별 분포", "PRD_DE": "201906", "TBL_ID": "DT_1B040A3", "ITM_NM": "제조업", "ITM_NM_ENG": "Koreans (Total)", "ITM_ID": "T20", "UNIT_NM": "천원", "ORG_ID": "101", "UNIT_NM_ENG": "Person", "C1_OBJ_NM": "월별총생산액(제조업)의 지역별 분포", "C1_OBJ_NM_ENG": "By Administrative District", "DT": "2957024", "PRD_SE": "M", "C1": "28", "C1_NM": "인천광역시", "C1_NM_ENG": "Incheon"}, {"TBL_NM": "월별총생산액(제조업)의 지역별 분포", "PRD_DE": "201906", "TBL_ID": "DT_1B040A3", "ITM_NM": "제조업", "ITM_NM_ENG": "Koreans (Total)", "ITM_ID": "T20", "UNIT_NM": "천원", "ORG_ID": "101", "UNIT_NM_ENG": "Person", "C1_OBJ_NM": "월별총생산액(제조업)의 지역별 분포", "C1_OBJ_NM_ENG": "By Administrative District", "DT": "1459024", "PRD_SE": "M", "C1": "29", "C1_NM": "대전광역시", "C1_NM_ENG": "Daejeon"}]
```

2. 통계청 포털 API 데이터 불러오기 (JSON)

6. Python 활용 불러오기

Requests 활용하여 url 내 제공된
데이터 획득가능

라이브러리 선언

```
import pandas as pd
import requests
from pandas.io.json import json_normalize
```

데이터 접속 주소 입력

url = '통계청 API에서 받은 URL'

정상 여부 확인 (200 정상)

response = requests.get(url)

response

JSON 데이터 획득

json = response.json()

PandasDataframe변환

df = json_normalize(json)

df.head()

```
import pandas as pd
import requests
from pandas.io.json import json_normalize
```

데이터 접속 주소 입력

url = 'http://kosis.kr/openapi/Param/statisticsParameterData.do?method=getList&apiKey=MjUzMtDiMjcyZjQwY2VkMWJkYjY='

정상 여부 확인 (200 정상)

response = requests.get(url)

response

JSON 데이터 획득

json = response.json()

PandasDataframe변환

df = json_normalize(json)

df.head()

C1	C1_NM	C1_NM_ENG	C1_OBJ_NM	C1_OBJ_NM_ENG	C2	C2_NM	C2_NM_ENG	C2_OBJ_NM	C2_OBJ_NM_ENG	...
----	-------	-----------	-----------	---------------	----	-------	-----------	-----------	---------------	-----

0	0	수주총액	Total amount of orders received	발주자별	By the kind of orderer	1	건축	Building	공종별	By the kind of construction works	...
---	---	------	---------------------------------	------	------------------------	---	----	----------	-----	-----------------------------------	-----

json 형태는 바로 변환 가능

지가변동률(시도/시/군/구) * 서울시 2010~2018년도 지가변동률 수집

기관명	통계표명	통계표ID	통계조사명	수록기간
통계청 (101)	지가변동률(시도/시/군/구)	DT_1YL20881	e-지방지표	년 2000~2018

URL생성 조건 설정

통계표명	지가변동률(시도/시/군/구)	통계표ID	DT_1YL20881
조회구분 ?	시계열		

URL생성 상세조건 정보

출력형태 설정	
데이터 포맷	<input checked="" type="radio"/> JSON <input type="radio"/> SDMX
조회기간 설정 <input checked="" type="radio"/> 기간설정 <input type="radio"/> 최근시점기준	
기간설정	주기 <input type="text" value="년"/> 시점 <input type="text" value="2016"/> ~ <input type="text" value="2018"/>

	C1	C1_NM	C1_NM_ENG	C1_OBJ_NM	DT	ITM_ID	ITM_NM	ORG_ID	PRD_DE
0	11	서울특별시	Seoul	행정구역별	0.527	T10	지가변동률	101	2010
1	11	서울특별시	Seoul	행정구역별	0.966	T10	지가변동률	101	2011
2	11	서울특별시	Seoul	행정구역별	0.378	T10	지가변동률	101	2012
3	11	서울특별시	Seoul	행정구역별	1.206	T10	지가변동률	101	2013
4	11	서울특별시	Seoul	행정구역별	2.662	T10	지가변동률	101	2014
5	11	서울특별시	Seoul	행정구역별	2.688	T10	지가변동률	101	2015
6	11	서울특별시	Seoul	행정구역별	2.974	T10	지가변동률	101	2016
7	11	서울특별시	Seoul	행정구역별	4.315	T10	지가변동률	101	2017
8	11	서울특별시	Seoul	행정구역별	6.114	T10	지가변동률	101	2018

3. 공공데이터 포털 API 데이터 불러오기

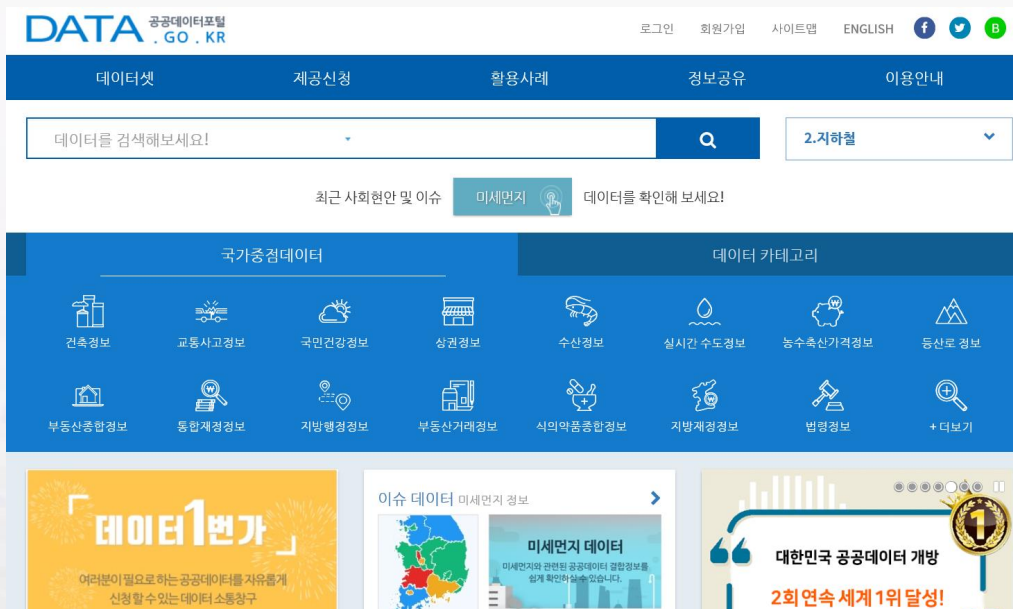
3. 공공데이터 포털 API 데이터 불러오기 (XML)

XML 데이터 불러오는 순서

- 1 접속 및 회원가입
- 2 자료검색 및 제공유형 확인
- 3 데이터 활용신청
- 4 개발계정 API 키 획득
- 5 데이터 인증 URL 획득
- 6 XML 태그 이해
- 7 Python 활용 불러오기

3. 공공데이터 포털 API 데이터 불러오기 (XML)

접속: (<https://www.data.go.kr/>) 및 회원가입



3. 공공데이터 포털 API 데이터 불러오기 (XML)

2. 자료검색 및 제공유형 확인

검색어로 데이터 탐색

DATA 공공데이터포털 .GO.KR 마이페이지 로그아웃 사이트맵

데이터셋 제공신청 활용사례 정보공유

부동산 거래 정보

연관 검색어 아파트 전월세 자료

1. 자료 검색

전체(16) 파일데이터(14) 오픈API(2) 표준데이터(0)

오픈API 2건을 찾았습니다.

기관별 검색

오픈API [2건]

정확도 ▾

날 짜

제 목

조회수

활용신청

국토교통부 실거래가 정보

조회수 : 365 활용신청건수 : 12,040

수정일 : 2018.12.04 기관 : 국토교통부 서비스유형 : REST

부동산 거래신고에 관한 법률에 따라 신고된 주택의 실거래 자료를 제공

XML

2. API 제공여부 확인

오픈API [1건]

정확도 ▾

날 짜

제 목

조회수

활용신청

국토교통부 실거래가 정보

조회수 : 365 활용신청건수 : 12,040

수정일 : 2018.12.04 기관 : 국토교통부 서비스유형 : REST

부동산 거래신고에 관한 법률에 따라 신고된 주택의 실거래 자료를 제공

XML

공공데이터

3. 공공데이터 포털 API 데이터 불러오기 (XML)

3. 데이터 활용신청

활용신청 시 로그인 요청

국토교통부 실거래가 정보

ENGLISH

부동산 거래신고에 관한 법률에 따라 신고된 주택의 실거래 자료를 제공

활용신청 (비로그인) 건수 : 12,026

1. 활용신청

활용신청 시 로그인 요청

XML

아파트매매 실거래 상세 자료

개발자네트워크

활용신청

단기

오류신고

★

XML

상업업무용 부동산 매매 신고 자료

개발자네트워크

활용신청

상세정보

오류신고

★

아파트매매 실거래 상세 자료

End Point

http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev?_wadl&type=xml

스크립

데이터 포맷

XML

API 유형

REST

비용부 과유무

무료

활용신청 건수

1873

심의유형

개발계정: 자동승인 / 운영계정: 자동승인

등록일

2017-02-06

수정일

2018-1

이용허락범위

이용허락범위 제한 없음

참고문서

국토교통부 실거래가 오픈API 활용가이드.hwp

미리보기

관리부서명

토지정책과

관리부서 전화번호

044-201-3407

* 자세한 데이터 설명자료 다운 가능

3. 공공데이터 포털 API 데이터 불러오기 (XML)

3. 데이터 활용신청

시스템유형, 정보, 라인센스 동의 후 신청

OPEN API > 개발계정 신청

개발계정 신청

기본정보

서비스명	아파트매매 실거래 상세 자료	서비스 유형	REST
심의여부	자동승인	신청유형	개발계정 변경신청
처리상태	신청	활용기간	승인일로부터 24개월 간 활용가능

안내

2014년 11월 10일 부터 API 서비스 인증키 발급 방식이 변경되었습니다. 프로그램을 통한 발급방식(구, 서버유형)은 폐지되었으며, 포털직접발급 방식(구, 모바일 유형)으로 통합되었습니다. 기존에 발급 받으신 서버용 인증키는 활용기간 까지 사용가능하며, 인증키를 발급 또는 재발급 받으시려면 해당 API서비스 계정 상세페이지 상단의 서버용 인증키발급(재발급) 버튼을 클릭 하시길 바랍니다.

시스템유형 선택

시스템유형

일반

서버 구축

* 일반 : OpenAPI 서비스를 호출하여 응답받은 결과값

* 서버 구축 : OpenAPI 서비스를 호출하여 응답받은 결과값

활용정보

활용목적

웹사이트 개발

앱개발 (모바일,솔루션등)

기타

참고자료

연구(논문 등)

첨부파일

※파일 첨부시 팝업창단 기능이 해제되어야 합니다.

추가

삭제

한 개의 파일만 첨부 할 수 있습니다.

상세기능정보 필수 입력 정보입니다.

*자동승인 상세기능은 신청과 동시에 활용 가능합니다.

	상세기능	설명	일일 트래픽
<input type="checkbox"/>			
<input checked="" type="checkbox"/>	아파트매매 상세자료	지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료 를 제공하는 아파트 매매 신고 정보 조회	1000

라이선스표시

이용허락범위

이용허락범위 제한 없음

(사유:) ☐ 동의합니다

신청

취소

System Message

개발계정 신청이 완료되었습니다.
'마이페이지>OPEN API>개발계정'에서 신청내역을 확인하실 수 있습니다.

확인

1. 시스템 유형: 일반

2. 활용자료: 기타

2. 체크 및 신청

- 데이터 활용 신청완료 후 1시간~하루 후 사용가능하다.
- * url을 웹페이지 열고 확인시 20 코드 에러 발생 시 기다려야함

3. 공공데이터 포털 API 데이터 불러오기 (XML)

4. 개발계정 API 키 획득

마이페이지 → 오픈API → 개발계정 API키 받기

홈 > 마이페이지 > OPEN API > 개발계정

개발계정

활용자가 개발계정을 신청중인 단계			활용자의 개발계정신청이 승인되어 활용중인 단계			활용자가 중지신청하여 운영이 중지된 단계		
신청			활용			중지		
0건			4건			0건		
· 반려 0건			· 변경신청 0건					
· 보류 0건			· 반려 0건					
			· 보류 0건					

상세검색

총 4건

[승인] 공동주택가격정보서비스

신청일:2019-04-17 [활용신청] 만료예정 :2021-04-17

서비스유형 :REST 분류 :지역개발 > 지역및도시 제공기관 :국토교통부

[승인] 대기오염정보 조회 서비스

신청일:2019-04-17 [변경신청] 만료예정 :2021-04-17

서비스유형 :REST 분류 :환경보호 > 대기 제공기관 :한국환경공단

[승인] 온라인 가격 정보

신청일:2019-04-17 [활용신청] 만료예정 :2021-04-17

서비스유형 :REST 분류 :일반공공행정 > 재정·금융 제공기관 :통계청

[승인] 아파트매매 실거래 상세 자료

신청일:2019-04-17 [활용신청] 만료예정 :2021-04-17

서비스유형 :REST 분류 :일반공공행정 > 일반행정 제공기관 :국토교통부

운영계정 신청

연장 신청

변경 신청

중지 신청

일반 인증키 받기

목록

기본정보

서비스명	온라인 가격 정보		상세설명		
서비스 유형	REST	일일트래픽	0	평균응답속도(초)	0
심의여부	자동승인	신청유형	개발계정 활용신청	처리상태	승인
활용기간	2019-04-17 ~ 2021-04-17				

서비스정보

일반 인증키 (UTF-8)	
End Point	http://data.insight.go.kr:8080/openapi/service/PriceInfo/getPriceInfo
데이터포맷	XML
참고문서	별첨_OpenAPI활용가이드_빅데이터공통기반_가격정보_v1.0.docx

3. 공공데이터 포털 API 데이터 불러오기 (XML)

5. 데이터 인증 URL 획득

마이페이지 → 오픈API → 개발계정 상세보기

홈 > 마이페이지 > 오픈API > 개발계정 상세보기

개발계정 상세보기

▶ 운영계정 신청 ▶ 연강 신청 ▶ 변경 신청 ▶ 중지 신청 ▶ 일반 인증키 재발급 ▶ 목록

▶ 기본정보

서비스명	아파트매매 실거래 상세 자료 상세설명				
서비스 유형	REST	일일트래픽	0	평균응답속도(초)	0
심의여부	자동승인	신청유형	개발계정 활용신청	처리상태	승인
활용기간	2019-04-17 ~ 2021-04-17				

▶ 서비스정보

일반 인증키 (UTF-8)	AT33s775KYpJOkUBJu0dxkJuUelfDIOJRzAH084EQS3JN%2BzFjErLHuk%2FGZa9L4gBTSGCzeA69tI9PwLp7B37IQ%3D%3D 복사
End Point	http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBSvc/getRTMSDataSvcAptTradeDev?_wadl&type=xml
데이터포맷	XML
참고문서	국토교통부 실거래가 오픈API 활용가이드.hwp

1. 참고문서

xmlurl="http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBSvc/getRTMSDataSvcAptTradeDev?LAWD_CD=11110&DEAL_YMD=201512&serviceKey=AT33s775KYpJOkUBJu0dxkJuUelfDIOJRzAH084EQS3JN%2BzFjErLHuk%2FGZa9L4gBTSGCzeA69tI9PwLp7B37IQ%3D%3D"

○ 응답메시지 예제

REST(URI)

http://openapi.molit.go.kr:8081/OpenAPI_ToolInstallPackage/service/rest/RTMSOBSvc/getRTMSDataSvcAptTradeDev?LAWD_CD=11110&DEAL_YMD=201512&serviceKey=서비스키

응답 메시지

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<response>
<header>
<resultCode>00</resultCode>
<resultMsg>NORMAL SERVICE.</resultMsg>
</header>
<body>
```

2. 인증 URL 경로 획득

• 데이터 종류에 따라 다수의 데이터가 존재할 수 있음
상품목록, 가격목록 2개면 URL이 2개 존재한다. 확인 必

3. 공공데이터 포털 API 데이터 불러오기 (XML)

6. XML 태그 이해

마이페이지 → 오픈API → 개발계정 상세보기

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <header>
    <resultCode>00</resultCode>
    <resultMsg>NORMAL SERVICE.</resultMsg>
  </header>
  <body>
    <items>
```

body → items → item 반복

```
      <item>
        <거래금액>82,500</거래금액>
        <건축년도>2008</건축년도>
        <년>2015</년>
        <도로명>사직로8길</도로명>
        <도로명건물부번호코드>00004</도로명건물부번호코드>
        <도로명건물부번호코드>00000</도로명건물부번호코드>
        <도로명시군구코드>11110</도로명시군구코드>
        <도로명일련번호코드>03</도로명일련번호코드>
        <도로명지상지하코드>0</도로명지상지하코드>
        <도로명코드>4100135</도로명코드>
        <법정동>사직동</법정동>
        <법정동부번호코드>0009</법정동부번호코드>
        <법정동부번호코드>0000</법정동부번호코드>
        <법정동시군구코드>11110</법정동시군구코드>
        <법정동읍면동코드>11500</법정동읍면동코드>
        <법정동지번코드>1</법정동지번코드>
        <아파트>광화문푸림스페이스본(101동-105동)</아파트>
        <월>12</월>
        <일>1-10</일>
        <일련번호>11110-2203</일련번호>
        <전용면적>94.51</전용면적>
        <지번>9</지번>
        <지역코드>11110</지역코드>
        <층>11</층>
      </item>
    </items>
  </body>
</response>
```

body

items

item

item

item

1행 데이터

2행 데이터

3행 데이터

3. 공공데이터 포털 API 데이터 불러오기 (XML)

6. XML 태그 이해

거래금액	건축년도	...
82,500	2008	...
60,000	1981	...

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <header>
    <resultCode>00</resultCode>
    <resultMsg>NORMAL SERVICE.</resultMsg>
  </header>
  <body>
    <items>
      <item>
        <거래금액> 82,500</거래금액>
        <건축년도>2008</건축년도>
      </item>
      <item>
        <거래금액> 60,000</거래금액>
        <건축년도>1981</건축년도>
      </item>
    </items>
  </body>
</response>
```



전체 row를 데이터 담을 리스트!



각 행별 컬럼값 담을 리스트

1행 값 담을 리스트

2행 값 담을 리스트

3행 값 담을 리스트

```
rowList = []
columnList = []
```

```
columnList.append("82,500")
columnList.append("2008")
rowList.append(columnList)
columnList = []
```

```
columnList.append("60,000")
columnList.append("1982")
rowList.append(columnList)
columnList = []
```

```
pd.DataFrame(rowList)
```


3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (Part #1)

#Used to service API connection

import requests

#Used to parse XML

from lxml import html

from bs4 import BeautifulSoup

import pandas as pd

url= '크롤링 대상 url'

response = requests.get(url)

Convert webpage to %Data

Data = BeautifulSoup(response.text, 'lxml-xml')

```
'<?xml version="1.0" encoding="UTF-8" standalone="yes"?><response><header><resultCode>00</resultCode><resultMsg>NORMAL SERVICE.</resultMsg>
</header><body><items><item><거래금액> 82,500</거래금액><건축년도>2008</건축년도><년>2015</년><도로명>사직로8길</도로명><도로명건물본번호
코드>00004</도로명건물본번호코드><도로명건물부번호코드>00000</도로명건물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일련번
코드>03</도로명일련번호코드><도로명지상지하코드>0</도로명지상지하코드><도로명코드>4100135</도로명코드><법정동> 사직동</법정동><법정동본번코
드>0009</법정동본번코드><법정동부번코드>0000</법정동부번코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드>11500</법정동읍면동
코드><법정동지번코드>1</법정동지번코드><아파트>광화문풍림스페이스본(101동-105동)</아파트><월>12</월><일>1-10</일><일련번호>11110-2203</일련
번호><전용면적>94.51</전용면적><지번>9</지번><지역코드>11110</지역코드><층>11</층></item><item><거래금액> 60,000</거래금액><건축년도>1981
</건축년도><년>2015</년><도로명>세종대로23길</도로명><도로명건물본번호코드>00047</도로명건물본번호코드><도로명건물부번호코드>00000</도로명건
물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일련번호코드>02</도로명일련번호코드><도로명지상지하코드>0</도로명지상지하코드>
<도로명코드>4100190</도로명코드><법정동> 당주동</법정동><법정동본번코드>0145</법정동본번코드><법정동부번코드>0000</법정동부번코드><법정동시
군구코드>11110</법정동시군구코드><법정동읍면동코드>11700</법정동읍면동코드><법정동지번코드>1</법정동지번코드><아파트>롯데미도파광화문빌딩</
아파트><월>12</월><일>21-31</일><일련번호>11110-12</일련번호><전용면적>149.95</전용면적><지번>145</지번><지역코드>11110</지역코드><층>8</층>
</item><item><거래금액> 130,000</거래금액><건축년도>2004</건축년도><년>2015</년><도로명>경희궁2길</도로명><도로명건물본번호코드>00005</도
로명건물본번호코드><도로명건물부번호코드>00005</도로명건물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일련번호코드>01</도로
명일련번호코드><도로명지상지하코드>0</도로명지상지하코드><도로명코드>4100005</도로명코드><법정동> 내수동</법정동><법정동본번코드>0110</법정
동본번코드><법정동부번코드>0015</법정동부번코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드>11800</법정동읍면동코드><법정동
읍면동지번코드>1</법정동읍면동지번코드><아파트>경희궁2길</아파트><월>12</월><일>1-10</일><일련번호>11110-2203</일련번호><전용면적>94.51</전용면적><지번>9</지번><지역코드>11110</지역코드><층>11</층></item></items></body></response>
```

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (Part #2)

전체 Row를 담은 리스트

dataframeList = []

한 행의 컬럼값을 담은 리스트

columnList = []

행 값을 담고 있는 모든 태그를 검색

rows = Data.find_all('item')

iterData는 ResultSet으로
리스트와 같이 [0] 등으로 접근가능하다.

1 iterData

```
[<item><거래금액> 82,500</거래금액><건축년도>2008</건축년도><년>2015</년><도로명>사직로8길</도로명><도로명건물본번호코드>00004</도로명건물본번호코드><도로명건물부번호코드>00000</도로명건물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일련번호코드>03</도로명일련번호코드><도로명지상지하코드>0</도로명지상지하코드><도로명코드>4100135</도로명코드><법정동> 사직동</법정동><법정동본번호코드>0009</법정동본번호코드><법정동부번호코드>0000</법정동부번호코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드>11500</법정동읍면동코드><법정동지번호코드>1</법정동지번호코드><아파트>광화문풍림스페이스본(101동~105동)</아파트><월>12</월><일>10</일><일련번호>11110-2203</일련번호><전용면적>94.5</전용면적><지번>9</지번><지역코드>11110</지역코드><층>11</층></item>
<item><거래금액> 60,000</거래금액><건축년도>1981</건축년도><년>2015</년><도로명>세종대로23길</도로명><도로명건물본번호코드>00047</도로명건물본번호코드><도로명건물부번호코드>00000</도로명건물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일련번호코드>02</도로명일련번호코드><도로명지상지하코드>0</도로명지상지하코드><도로명코드>4100190</도로명코드><법정동> 당주동</법정동><법정동본번호코드>0145</법정동본번호코드><법정동부번호코드>0000</법정동부번호코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드>11700</법정동읍면동코드><법정동지번호코드>1</법정동지번호코드><아파트>롯데미도파광화문빌딩</아파트><월>12</월><일>22</일><일련번호>11110-12</일련번호><전용면적>149.95</전용면적><지번>145</지번><지역코드>11110</지역코드><층>8</층></item>
```

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (Part #3)

한개 행의 모든컬럼 값을 리스트에 담아보자!

```
columns = rows[0].find_all()
```

거래금액

```
eachcolumnValue = columns[0].text  
columnList.append(eachcolumnValue)
```

건축년도

```
eachcolumnValue = columns[1].text  
columnList.append(eachcolumnValue)
```

....

반복문으로 변환

```
for j in range(0, len(columns)):  
    eachcolumnValue = columns[j].text  
    columnList.append(eachcolumnValue)
```

1	rows[0].find_all()
[<거래금액> 82,500</거래금액>, <건축년도>2008</건축년도>, <년>2015</년>, <도로명>사직로8길</도로명>, <도로명건물본번호코드>00004</도로명건물본번호코드>, <도로명건물부번호코드>00000</도로명건물부번호코드>, <도로명시군구코드>11110</도로명시군구코드>, <도로명일련번호코드>03</도로명일련번호코드>, <도로명지상지하코드>0</도로명지상지하코드>, <도로명코드>4100135</도로명코드>, <법정동> 사직동</법정동>, <법정동본번호코드>0009</법정동본번호코드>, <법정동부번호코드>0000</법정동부번호코드>, <법정동시군구코드>11110</법정동시군구코드>, <법정동읍면동코드>11500</법정동읍면동코드>, <법정동지번코드>1</법정동지번코드>, <아파트>광화문풍림스페이스본(101동~105동)</아파트>, <월>12</월>, <일>10</일>, <일련번호>11110-2203</일련번호>, <전용면적>94.51</전용면적>, <지번>9</지번>, <지역코드>11110</지역코드>, <층>11</층>]	

1	columnList
[' 82,500', '2008', '2015', '사직로8길', '00004', '00000', '11110', '03', '0', '4100135', ' 사직동', '0009', '0000', '11110', '11500', '1', '광화문풍림스페이스본(101동~105동)', '12', '10', '11110-2203', '94.51', '9', '11110', '11']	

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (Part #4)

row를 반복한다.

```
for i in range(0, len(rows)):
    columns = rows[i].find_all()
    for j in range(0, len(columns)):
        eachcolumnValue = columns[j].text
        columnList.append(eachcolumnValue)
    # 데이터프레임 리스트에 컬럼값을 담자!
    # list( (1행컬럼값), (2행컬럼값), (3행컬럼값),... )
    dataframeList.append(columnList)
    # 컬럼값을 다시 담기 위해 리스트를 비운다.
    columnList = []
```

dataframeList

1 dataframeList

```
[[ '      82,500',
  '2008',
  '2015',
  '사직로8길',
  '00004',
  '00000',
  '11110',
  '03',
  '0',
  '4100135',
  '사직동',
  '0009',
  '0000',
  '11110',
  '11500',
  '1',
```

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (Part #5)

데이터 프레임으로 변환하자
`pd.DataFrame(dataframeList)`

	0	1	2	3	4	5	6	7	8	9	...	14	15
0	82,500	2008	2015	사직로8 길	00004	00000	11110	03	0	4100135	...	11500	1

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (리팩토링)

```
for i in range(0, len(rows)):
    columns = rows[i].find_all()
    for j in range(0, len(columns)):
        eachcolumnValue = columns[j].text
        columnList.append(eachcolumnValue)
    dataframeList.append(columnList)
    columnList = []
```



rows를 한개씩 꺼내서
eachrow 변수에 담는다.

```
for eachrow in rows:
    columns = eachrow.find_all()
    for eachColumns in columns:
        eachcolumnValue = eachColumns.text
        columnList.append(eachcolumnValue)
    dataframeList.append(columnList)
    columnList = []
```

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (컬럼이름 추출)

```
# 전체 Row를 담은 리스트  
dataframeList = []
```

```
# 한 행의 컬럼값을 담은 리스트  
columnList = []
```

```
# 행 값을 담고 있는 모든 태그를 검색  
rows = Data.find_all('item')
```

```
# 첫번째 행때 컬럼 수집  
rowcnt = 0  
columnNameList = []
```

```
for eachrow in rows:
```

```
    columns = eachrow.find_all()
```

```
    for eachColumns in columns:
```

```
        if(rowcnt==0):
```

```
            columnNameList.append(eachColumns.name)
```

```
            eachcolumnValue = eachColumns.text
```

```
            columnList.append(eachcolumnValue)
```

```
        rowcnt=rowcnt+1
```

```
    dataframeList.append(columnList)
```

```
    columnList = []
```

```
# 데이터 프레임으로 변환하자
```

```
result2 = pd.DataFrame(dataframeList)
```

3. 공공데이터 포털 API 데이터 불러오기 (XML)

7. Python 활용 불러오기 (컬럼이름 매핑)

```
result2.columns = columnNameList  
result2
```

1	result2												
	거래금 액	건축 년도	년	도로 명	도로명건 물본번호 코드	도로명건 물부번호 코드	도로명 시군구 코드	도로명 일련번 호코드	도로명 지상지 하코드	도로명 코드	...	법정동 읍면동 코드	법정 동지 번코 드
0	82,500	2008	2015	사직 로8 길	00004	00000	11110	03	0	4100135	...	11500	1
1	60,000	1981	2015	세종 대로 23 길	00047	00000	11110	02	0	4100190	...	11700	1

대기오염정보 검색 후 통합대기환경지수 나쁨 이상 측정소 목록조회

오픈API [1건] 정확도 ▾ 날짜 제목 조회수 활용신청

한국환경공단_대기오염정보 조회수 : 5,144 활용신청건수 : 16,572

수정일 : 2019.05.24 기관 : 한국환경공단 서비스유형 : REST

..소별 대기오염정보를 조회하기 위한 서비스로 기간별, 시도별 대기오염 정보와 통합대기환경지수 나쁨 이상 측정소 내역, 대...

XML

환경기상

[승인] 대기오염정보 조회 서비스

신청일:2019-04-17

서비스유형 :REST

분류 :환경보호 > 대기

제공기관 :한국환경공단

1. 서비스 목록

순번	서비스 ID	서비스명(국문)	서비스명(영문)
1	SC-OA-09-01	측정소정보 조회 서비스	MsrstnInfolnquireSvc
2	SC-OA-09-02	대기오염정보 조회 서비스	ArpltnInforlnquireSvc
3	SC-OA-09-03	대기오염통계 서비스	ArpltnStatsSvc
4	SC-OA-09-04	오존황사 발생정보조회 서비스	OzYlwsndOccrrncInforlnquireSvc
5	SC-OA-09-05	미세먼지 경보 정보 조회 서비스	UlfptcaAlarmlnquireSvc

(2) 통합대기환경지수 나쁨 이상 측정소 목록조회 오퍼레이션 명세

오퍼레이션 정보	오퍼레이션 번호	2	오퍼레이션명(국문)	통합대기환경지수 나쁨 이상 측정소 목록조회
	오퍼레이션 유형	조회	오퍼레이션명(영문)	getUnityAirEnvrnIdxSnstiveAboveMsrstnList
	오퍼레이션 설명	통합대기환경지수가 나쁨 등급 이상인 측정소명과 주소 목록 정보를 제공하는 통합 대기환경지수 나쁨 이상 측정소 목록조회		
	URL end-point	http://openapi.airkorea.or.kr/openapi/services/rest/ArpltnInforlnquireSvc/getUnityAirEnvrnIdxSnstiveAboveMsrstnList		
	최대 메시지 사이즈	[1000K bytes]		
	평균 응답 시간	[500 ms]	초당 최대 트랜잭션	[30 tps]

<http://openapi.airkorea.or.kr/openapi/services/rest/ArpltnInforlnquireSvc/getUnityAirEnvrnIdxSnstiveAboveMsrstnList?pageNo=1&numOfRows=10&ServiceKey=AT33s775KYpJOkUBJu0dxkJuUelfDIOJRzAH084EQS3JN%2BzFjErLHuk%2FGZa9L4gBTSGCzeA69tI9PwLp7B37IQ%3D%3D>

* 자세한 데이터
설명자료 다운
가능

4. 핵심정리 및 Q&A

기억합시다

1

공공데이터 포털 등 공개정보에 접근하여 분석할 수 있는 방법 기억

2

API, 파일데이터 등의 종류로 개방되어 있으므로 불러오는 방법 기억

3

특정 데이터는 데이터 활용을 위해 기달려야 함을 기억

참조

Pdf 데이터 불러오기

pdf 데이터 불러오는 순서

- 1 데이터 확인
- 2 라이브러리 선언
- 3 pdf 데이터 불러오기
- 4 현재날짜 지정
- 5 데이터 저장하기


3. pdf 데이터 불러오기

접속: (http://fsc.go.kr/info/trd_list.jsp?menu=7230000&bbsid=BBS0069)

금융시장동향

· 홈 > 알림마당 > 금융시장동향



날짜검색 ~

조건검색 제목

Q 검색

전체 3694 페이지 1/247

번호	제목	등록일	조회	첨부파일
3694	금융시장동향(19. 7. 19)	2019-07-19	31	 
3693	금융시장동향(19. 7. 18)	2019-07-18	69	 
3692	금융시장동향(19. 7. 17)	2019-07-17	237	 
3691	금융시장동향(19. 7. 16)	2019-07-16	775	 
3690	금융시장동향(19. 7. 15)	2019-07-15	817	 
3689	금융시장동향(19. 7. 12)	2019-07-12	1373	 
3688	금융시장동향(19. 7. 11)	2019-07-11	94	 
3687	금융시장동향(19. 7. 10)	2019-07-10	268	 

2. pdf 데이터 불러오기

1. 데이터 확인

국내 금융시장 일일동향 (7.19일)

(금융시장분석과)

- ◆ 국내 금융시장은 향후 韓 통화정책 완화 기대감 및 美 기준금리 인하 기대 영향 등으로 주가 상승, 환율 하락, 금리 하락
 - 코스피, 코스닥 모두 뉴욕 연은 총재의 발언에 의한 기준금리 인하 기대 영향 등으로 각각 +1.35%, +1.34% 상승
 - 주세별 코스피 순매수(억원) : (외국인) +1,325 (기관) +3,373 (개인) △4,619
 - 주세별 코스닥 순매수(억원) : (외국인) △214 (기관) +187 (개인) +31
 - 원달러 환율은 달러가 약세를 보인 영향으로 △4.3원 하락
 - 국제금리는 기준금리 인하 발표 및 향후 통화정책 완화 기대감으로 하락(3년물 △1.7bp, 10년물 △1.6bp)

- (주가) 코스피는 뉴욕 연은 총재의 통화완화적 발언에 의한 금리인하 기대감이 증가한 영향 등으로 상승 <증권(+2.37%), 보험업(+2.33%) 등>
 - 코스닥은 외국인 및 기관 순매수 영향으로 5거래일 만에 반등

(단위:pt)	18년 말	7.12	7.18	7.19	전일비	전주비	18년 말비
KOSPI	2,041	2,087	2,067	2,094	+1.3%	+0.4%	+2.6%
KOSDAQ	676	681	665	674	+1.3%	△1.0%	△0.2%

- (환율) 원달러 환율은 뉴욕 연은 총재의 통화완화적 발언에 의한 美 달러 약세 및 수출업체의 결제물량 출회 영향으로 하락

(단위:원)	18년 말	7.12	7.18	7.19	전일비	전주비	18년 말비
원/달러	1,115.7	1,178.7	1,178.5	1,174.5	△0.3%	△0.4%	+5.3%
원/100엔	1,017.1	1,087.9	1,094.1	1,090.9	△0.3%	+0.3%	+7.3%

- (금리) 국제금리는 한국은행의 기준금리 인하 발표 및 향후 통화정책 완화 기대감 등으로 하락

(단위:%)	18년 말	7.12	7.18	7.19	전일비	전주비	18년 말비
CD91일	1.930	1.780	1.610	1.600	△1.0bp	△18.0bp	△33.0bp
국고3년	1.817	1.424	1.345	1.325	△1.7bp	△9.9bp	△49.2bp
국고10년	1.948	1.569	1.472	1.454	△1.6bp	△11.5bp	△49.4bp
회사채(AA-)	2.287	1.927	1.850	1.829	△2.1bp	△9.8bp	△45.8bp
장단기스프레드	13.1	14.5	12.7	12.9	+0.1bp	△1.6bp	△0.2bp
신용 스프레드	47.0	50.3	50.5	50.4	△0.4bp	+0.1bp	+3.4bp

주가

(단위:pt)	18년 말	7.12	7.18	7.19	전일비	전주비	18년 말비
KOSPI	2,041	2,087	2,067	2,094	+1.3%	+0.4%	+2.6%
KOSDAQ	676	681	665	674	+1.3%	△1.0%	△0.2%

환율

(단위:원)	18년 말	7.12	7.18	7.19	전일비	전주비	18년 말비
원/달러	1,115.7	1,178.7	1,178.5	1,174.5	△0.3%	△0.4%	+5.3%
원/100엔	1,017.1	1,087.9	1,094.1	1,090.9	△0.3%	+0.3%	+7.3%

금리

(단위:%)	18년 말	7.12	7.18	7.19	전일비	전주비	18년 말비
CD91일	1.930	1.780	1.610	1.600	△1.0bp	△18.0bp	△33.0bp
국고3년	1.817	1.424	1.345	1.325	△1.7bp	△9.9bp	△49.2bp
국고10년	1.948	1.569	1.472	1.454	△1.6bp	△11.5bp	△49.4bp
회사채(AA-)	2.287	1.927	1.850	1.829	△2.1bp	△9.8bp	△45.8bp
장단기스프레드	13.1	14.5	12.7	12.9	+0.1bp	△1.6bp	△0.2bp
신용 스프레드	47.0	50.3	50.5	50.4	△0.4bp	+0.1bp	+3.4bp

2. pdf 데이터 불러오기

2. 라이브러리 선언

#table,표를 다룰 때 사용하는 모듈

```
import tabula
```

#pdf 에서 표를 읽을 때 필요한 모듈, wrapper.read_pdf 로 사용

```
from tabula import wrapper
```

```
pip install tabula-py
```

Tabula-py is a simple Python wrapper of tabula-java, which can read the table of PDF. You can read tables from PDF and convert into pandas' DataFrame. tabula-py also enables you to convert a PDF file into CSV/TSV/JSON file.

2. pdf 데이터 불러오기

3. pdf 데이터 불러오기

```
wrapper.read_pdf( " url 또는 파일위치",  
                  multiple_tables = True,  
                  pages = "all",  
                  pandas_options = {"header":0} )
```

실습

```
tables = wrapper.read_pdf("../dataset/190719_금융시장동향(오후).pdf",  
                           multiple_tables=True,  
                           pages="all",  
                           pandas_options={"header":0})
```

```
type(tables)  
len(tables)
```

1	tables[0]							
	(단위:pt)	18년말	7.12	7.18	7.19	전일비	전주비	18년말비
0	KOSPI	2,041	2,087	2,067	2,094	+1.3%	+0.4%	+2.6%
1	KOSDAQ	676	681	665	674	+1.3%	△1.0%	△0.2%

2. pdf 데이터 불러오기

4. 현재 날짜 불러오기

```
from datetime import datetime
today = datetime.today()
today.현재 날짜
```

실습

```
from datetime import datetime
```

```
today = datetime.today()
```

```
currentyear = today.year
```

```
currentmonth = today.month
```

```
currentday = today.day
```

```
currentdate = str(currentyear)+str(currentmonth)+str(currentday)
```

2. pdf 데이터 불러오기

5. pdf 파일 내 테이블 csv로 저장하기

```
for i in range(0, len(tables)):
    filename = currentdate + "_금융시장동향_" + str(i) + ".csv"
    tables[i].to_csv("./"+filename, index=False, encoding="ms949")
```

실 습

```
for i in range(0, len(tables)):
    filename = currentdate + "_금융시장동향_" + str(i) + ".csv"
    tables[i].to_csv("./"+filename, index=False, encoding="ms949")
```

감사합니다

