# 4. 데이터 크롤링 기본 (02. 오픈 API)

김 효 관

# 외부연동 데이터 수집

## 단원 개요

**[단원명]**

외부연동 데이터 수집

**[단원 소개]**

- 데이터 분석 시 회사 내부에서 접근할 수 있는데이터 (파일, 데이터베이스) 만으로는 한계가 있다. 외부에서 활용가능한 포털 내 데이터 나 공개한 데이터를 수집하는 방법을 익히고 다양한 각도로 분석 모델을 만들어가는 방법을 학습한다.

**[교육대상]**

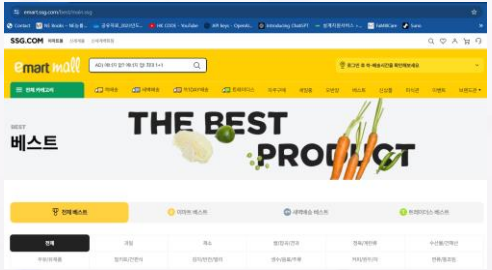
- 데이터 분석가 / 인공지능 전문가
- 데이터 엔지니어

| 내용 | 학습내용 |
|---|---|
| **웹 크롤링 활용 데이터 수집** | - 외부 데이터 수집의 필요성을 이해한다.<br>- 웹 포털 내 존재하는 데이터 수집 방법을 실습한다.<br>- 브라우저 자동화 라이브러리를 활용한 데이터 수집 자동화 방법을 실습한다. |
| **공공데이터 포털 데이터 수집** | - 공공데이터 포털을 이해한다.<br>- 공개된 파일 형태의 자료 수집 방법을 실습한다.<br>- API 형태로 공개한 JSON 포맷 데이터 수집 방법을 실습한다.<br>- API 형태로 공개한 XML 포맷 데이터 수집 방법을 실습한다. |

# 웹 크롤링

## 단원 개요

### 웹 크롤링




| | | 0 | 1 |
|---|---|---|---|
| 0 | 더마시나 무항생제 구운계란60구, 1개, 2,100g | 10,300 |
| 1 | 오복유통 HACCP인증 구운계란 2판60구, 60구, 2판 | 13,900 |
| 2 | 참나무촌 무염훈제계란, 30개입, 1.2kg(한판) | 6,200 |
| 3 | 꾼란 맥반석 구운계란 30구 1판, 30개입, 1.2kg | 7,900 |
| 4 | 맛군 촉촉 쫄깃 구운 계란, 30알, 1박스 | 7,900 |
| 5 | 감동란 간이베어 있는 촉촉한 반숙계란, 50g, 30개입 | 16,900 |
| 6 | [ 계란사랑 ] 맥반석 구운계란 구운란 60구 (2판), 2700g | 11,900 |
| 7 | 진주팸 오마이 포켓 메추리알 5p, 25g, 10개입 | 9,440 |

### 공공데이터 수집

| | 연도 | 월 | 전국 PIR | 서울 PIR | 부산 PIR | 대구 PIR | 인천 PIR | 광주 PIR | 대전 PIR | 울산 PIR |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2004 | 3 | 4.21 | 4.89 | 3.95 | 3.73 | 4.65 | 2.81 | 4.68 | 2.66 |
| 1 | 2004 | 4 | 4.39 | 5.59 | 3.91 | 3.88 | 4.59 | 2.92 | 3.83 | 2.74 |
| 2 | 2004 | 5 | 4.19 | 5.14 | 4.90 | 3.83 | 4.78 | 3.41 | 4.19 | 2.93 |
| 3 | 2004 | 6 | 4.09 | 4.38 | 4.20 | 3.77 | 4.30 | 2.83 | 4.19 | 2.81 |

| | 대기오염 나쁨 위치 | 관측소위치 |
|---|---|---|
| 0 | 중구 | 서울특별시 중구 덕수궁길 15시청서소문별관 3동 |
| 1 | 청계천로 | 서울 중구 청계천로 184(청계천4가사거리 남강빌딩 앞) |
| 2 | 용산구 | 서울 용산구 한남대로 136서울특별시중부기술교육원 |
| 3 | 강변북로 | 서울 성동구 강변북로 257한강사업본부 옆 |
| 4 | 홍릉로 | 서울 동대문구 홍릉로 1(청량리전철역 사거리 SC제일은행 앞) |

교육목표: Python을 활용하여 공개된 데이터를 불러올 수 있다.

# CONTENTS


1 공공데이터 포털 파일 데이터 불러오기

2 국가통계포털 API 데이터 불러오기

3 공공데이터 포털 API 데이터 불러오기

4 핵심정리 및 Q&A

# 1. 공공데이터 포털 파일 데이터 불러오기

## 다양한 공공 데이터 포털

다양한 공공 데이터 포털 데이터를 활용하면 실시간 업데이트된 데이터를  활용 가능

1. 공공데이터포털 파일 데이터 불러오기
2. 국가통계포털 API 데이터 불러오기
3. 공공데이터포털 API 데이터 불러오기

# 파트1. 공공데이터 포털 파일 데이터 불러오기

# 1. 공공데이터 포털 파일 데이터 불러오기

## 모듈 개요

**[과정개요]**

공공데이터 포털 파일 데이터 불러오기

**[교육목표]**

- 웹 상의 공개된 파일 데이터 접근방법 실습

**[교육대상]**

- 데이터 분석가 / 인공지능 전문가
- 데이터 엔지니어

| 내용 | 학습내용 |
|---|---|
| **공공데이터 포털 파일데이터 불러오기** | - 파일 형태로 공개한 포털 데이터를 불러오는 방법을 실습합니다. |

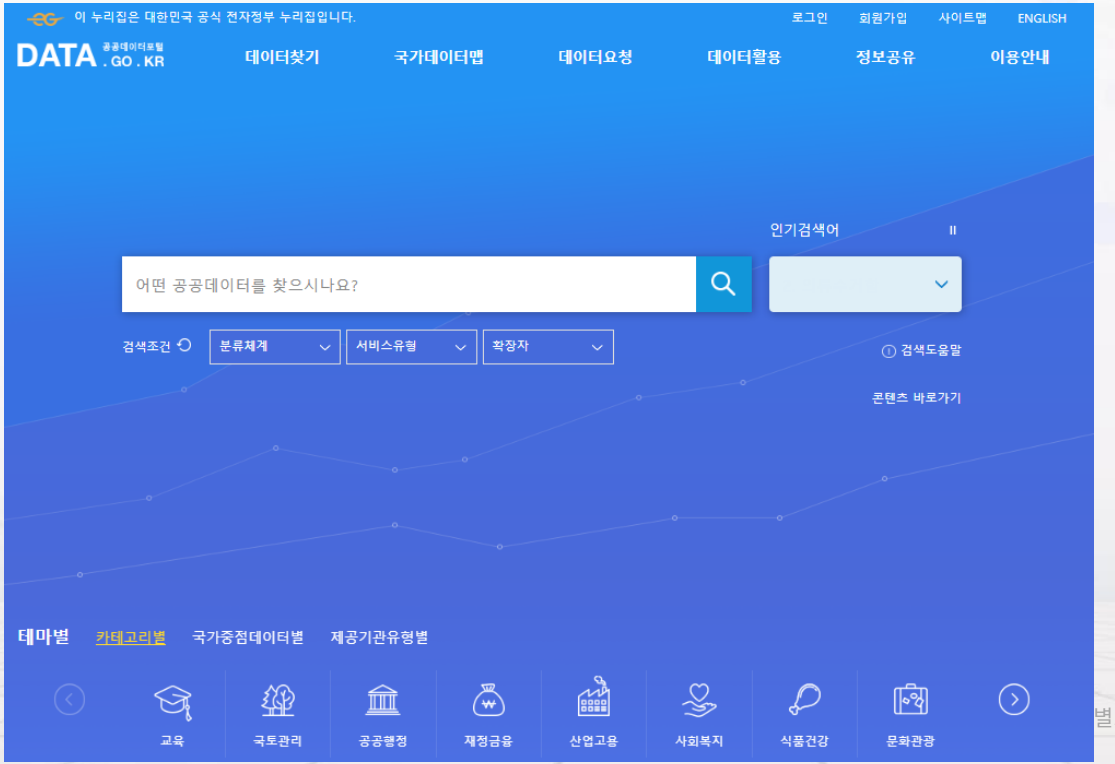
# 1. 공공데이터 포털 파일 데이터 불러오기

## 공공데이터 포털 파일 데이터 불러오기 절차

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

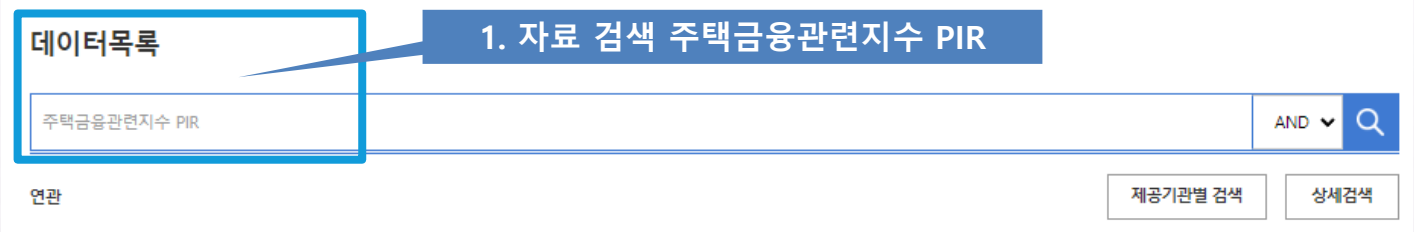
## 1. 접속: (https://www.data.go.kr/) 및 회원가입



# 1. 공공데이터 포털 파일 데이터 불러오기

## 2. 자료검색 및 제공유형 확인

### 검색어로 데이터 탐색



* PIR (Price to Income Ratio) : 연간소득대비 주택가격 비율
* LIR (Loan to Income Ratio) : 연간소득대비 대출금액 비율

# 1. 공공데이터 포털 파일 데이터 불러오기

## 3. 데이터 다운로드 url 확인

* PIR (Price to Income Ratio) : 연간소득대비 주택가격 비율
* LIR (Loan to Income Ratio) : 연간소득대비 대출금액 비율

**https://www.data.go.kr/cmm/cmm/fileDownload.do?atchFileId=FILE_000000002554243&fileDetailSn=1&dataNm=%ED%95%9C%EA%B5%AD%EC%A3%BC%ED%83%9D%EA%B8%88%EC%9C%B5%EA%B3%B5%EC%82%AC_%EC%A7%80%EC%97%AD%EB%B3%84%20PIR%20%EB%B0%8F%20LIR_20211231**

# 1. 공공데이터 포털 파일 데이터 불러오기

## 4. 파이썬 활용 데이터 불러오기

pd.read_csv(“url”)

```
import pandas as pd

url = "https://www.data.go.kr/cmm/cmm/fileDownload.do?atchFileId=FILE_000000002554243&fileDet

data = pd.read_csv(url,encoding="ms949")

data.tail()
```

| | 연도 | 월 | 전국소득대비주택가격비율 | 서울소득대비주택가격비율 | 부산소득대비주택가격비율 | 대구소득대비주택가격비율 | 인천소득대비주택가격비율 | 광주소득대비주택가격비율 | 대전소득대비주택가격비율 | 울산소득대비주택가격비율 | ... | 세종평균연소득대비평균대출금액 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **208** | 2021 | 7 | 6.98 | 9.11 | 7.68 | 6.82 | 7.65 | 6.44 | 7.25 | 5.52 | ... | 3.91 |
| **209** | 2021 | 8 | 6.74 | 8.41 | 7.32 | 7.08 | 7.37 | 6.73 | 7.73 | 5.99 | ... | 5.03 |
| **210** | 2021 | 9 | 6.70 | 8.76 | 7.75 | 6.99 | 7.83 | 6.56 | 7.14 | 6.08 | ... | 4.11 |
| **211** | 2021 | 10 | 6.69 | 8.55 | 6.91 | 7.67 | 8.92 | 7.17 | 6.72 | 6.65 | ... | 4.85 |
| **212** | 2021 | 11 | 7.10 | 0.00 | 10.75 | 15.29 | 12.03 | 7.47 | 7.11 | 7.42 | ... | 0.00 |

# 실습

## 검색어 “우편번호” -> 통계청_나라통계_우편번호

공공행정 | 국가행정기관

CSV | JSON + XML **통계청_나라통계_우편번호**

우편번호, 우편번호순서, 도이름, 시군구이름, 읍면동이름, 리건물이름, 동호, 최초등록일, 최초등록자, 최종수정일, 최종수정자, 등록일, 수정일시, 전체주소

| | 우편번호 | 우편번호순서 | 도이름 | 시군구이름 | 읍면동이름 | 리건물이름 | 동호 | 전체주소 |
|---|---|---|---|---|---|---|---|---|
| 0 | 137926 | 11 | 서울특별시 | 서초구 | 서초1동 | NaN | 1641-1~1641-13 | 서울특별시 서초구 서초1동 1641-1~1641-13 |
| 1 | 137927 | 11 | 서울특별시 | 서초구 | 서초3동 | NaN | 1451 | 서울특별시 서초구 서초3동 1451 |
| 2 | 137929 | 12 | 서울특별시 | 서초구 | 서초4동 | 서초교대e편한세상아파트 | (101~106동) | 서울특별시 서초구 서초4동 서초교대e편한세상아파트 (101~106동) |
| 3 | 137930 | 2 | 서울특별시 | 서초구 | 반포1동 | 반포자이아파트 | (101~144동) | 서울특별시 서초구 반포1동 반포자이아파트 (101~144동) |
| 4 | 137931 | 2 | 서울특별시 | 서초구 | 반포1동 | 삼호가든아파트 | (501~503동) | 서울특별시 서초구 반포1동 삼호가든아파트 (501~503동) |

https://www.data.go.kr/cmm/cmm/fileDownload.do?atchFileId=FILE_000000002336443&fileDetailSn=1&dataNm=%ED%86%B5%EA%B3%84%EC%B2%AD_%EB%82%98%EB%9D%BC%ED%86%B5%EA%B3%84_%EC%9A%B0%ED%8E%B8%EB%B2%88%ED%98%B8_20211110

# 실습

검색어 “상권정보” ->
소상공인시장진흥공단_상가(상권)정보

파일데이터 (20,507건)

산업고용 공공기관 국가중점

CSV,HWP,XLS,XLSX JSON + XML 소상공인시장진흥공단_상가(상권)정보

상권업종분류 : 표준산업분류 기반 업종분류 개편(837개 -> 247개)2. 표준산업분류 : 9차→10차3.

제공기관 소상공인시장진흥공단 수정일 2024-04-24 조회수 214457 다운로드 218354

https://www.data.go.kr/cmm/cmm/fileDownload.do?atchFileId=FILE_000000002336443&fileDetailSn=1&dataNm=%ED%86%B5%EA%B3%84%EC%B2%AD_%EB%82%98%EB%9D%BC%ED%86%B5%EA%B3%84_%EC%9A%B0%ED%8E%B8%EB%B2%88%ED%98%B8_20211110

# 파트2. 국가통계포털 API 데이터 불러오기

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 모듈 개요

**[과정개요]**

국가통계포털 API 데이터 불러오기(JSON)

**[교육목표]**

- 웹 상의 공개된 API 활용방법

**[교육대상]**

- 데이터 분석가 / 인공지능 전문가
- 데이터 엔지니어

| 내용 | 학습내용 |
|---|---|
| **국가통계포털 API 데이터 불러오기** | - API로 공개한 데이터의 수집방법을 실습합니다. (JSON 타입) |

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 국가통계포털 API 데이터 불러오기 절차

1. 국가통계포털 공유서비스 사이트 접속 및 회원가입
2. 검색 및 URL 생성
3. JSON 데이터 이해
4. Python 활용 불러오기

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 1. 국가통계포털 공유서비스 사이트 접속 및 회원가입

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

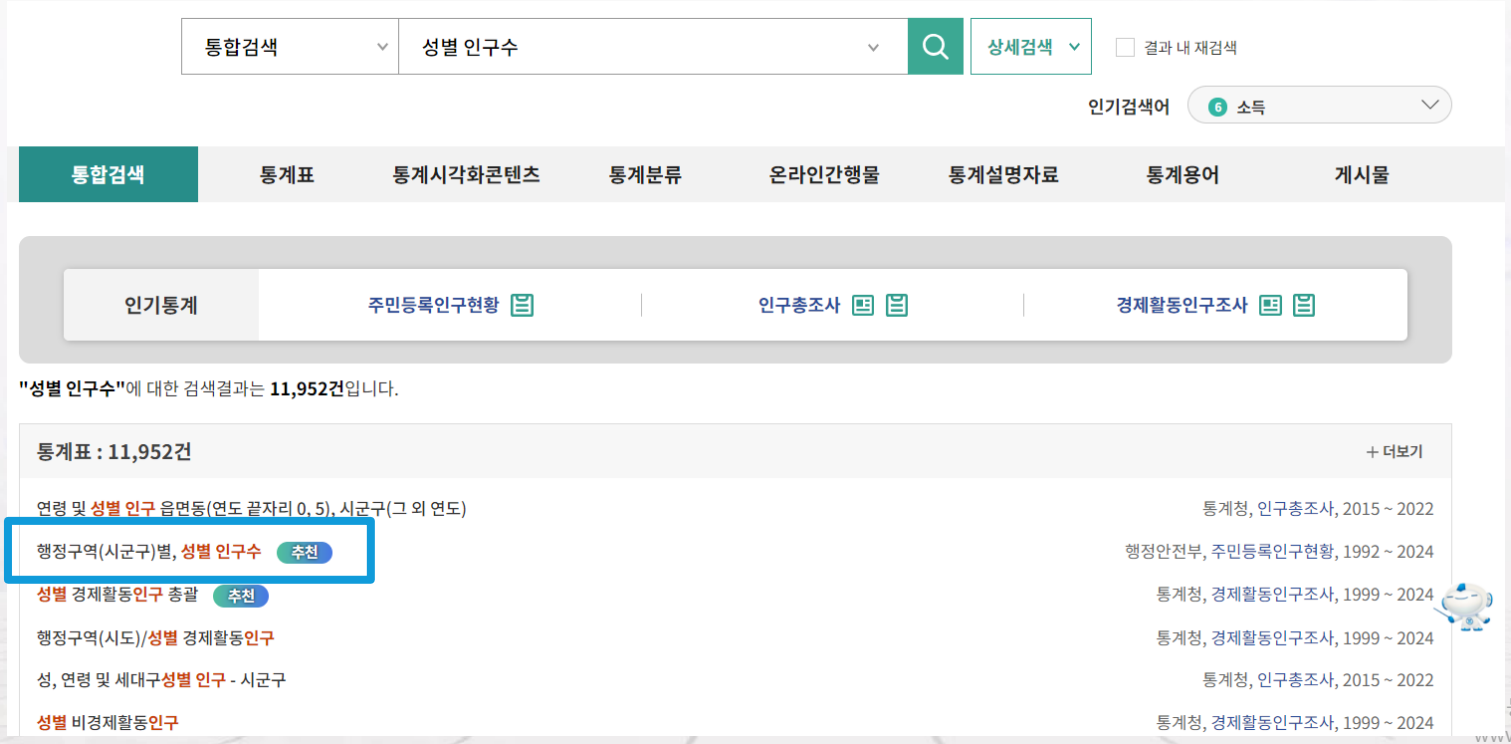
회원가입 후 활용 신청 진행!

행정구역별 성별 인구수 조회

| 접속 사이트 | 제공항목(데이터셋) | 폴더명 |
|---|---|---|
| 국가통계포털 | 행정구역(시군구)별, 성별 인구수 | 0_행정구역(시군구)별, 성별 인구수 |

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 2. 검색 및 URL 생성




능(광문각, 김효관 교수)
www.youtube.com/hkcode

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 2. 검색 및 URL 생성

1. 행정구역(시군구)별, 성별 인… | 2. 행정구역(시군구)별, 성별 … | 모두 닫기

1) 행정구역(시군구)별, 성별 인구수

「주민등록인구현황」 행정안전부 (자료문의처: 044-205-3158) 통계설명자료 | 온라인간행물

수록기간 : 월, 년 1992 ~ 2024.03 / 자료갱신일: 2024-04-03 / 주석정보

시점 | 증감/증감률 | 행렬전환 | 열고정해제 | 새 탭 열기 | 화면복사 | 주소/출처 | 스크랩 | OPENAPI | 인쇄 | 다운로드 | 조회설정

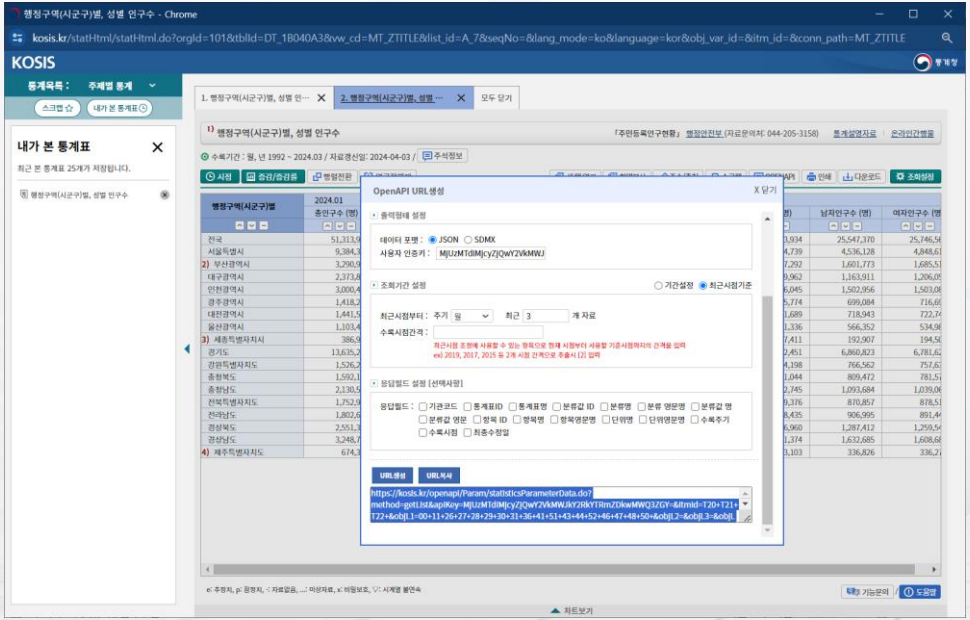
URL생성

1. 검색

| 행정구역(시군구)별 | 2024.01 | | | 2024.02 | | | 2024.03 |
|---|---|---|---|---|---|---|---|
| | 총인구수 (명) | 남자인구수 (명) | 여자인구수 (명) | 총인구수 (명) | 남자인구수 (명) | 여자인구수 (명) | |
| 전국 | 51,313,912 | 25,558,944 | 25,754,968 | | | | |
| 서울특별시 | 9,384,325 | 4,538,354 | 4,845,971 | | | | |
| 2) 부산광역시 | 3,290,964 | 1,603,859 | 1,687,105 | | | | |
| 대구광역시 | 2,373,844 | 1,166,110 | 1,207,734 | | | | |
| 인천광역시 | 3,000,454 | 1,500,414 | 1,500,040 | | | | |
| 광주광역시 | 1,418,241 | 700,377 | 717,864 | | | | |
| 대전광역시 | 1,441,562 | 719,008 | 722,554 | | | | |
| 울산광역시 | 1,103,402 | 567,081 | 536,321 | | | | |
| 3) 세종특별자치시 | 386,944 | 192,704 | 194,240 | | | | |
| 경기도 | 13,635,250 | 6,857,867 | 6,777,383 | | | | |
| 강원특별자치도 | 1,526,243 | 767,496 | 758,747 | | | | |
| 충청북도 | 1,592,155 | 809,830 | 782,325 | | | | |
| 충청남도 | 2,130,509 | 1,092,059 | 1,038,450 | | | | |
| 전북특별자치도 | 1,752,921 | 872,493 | 880,428 | | | | |
| 전라남도 | 1,802,672 | 908,705 | 893,967 | | | | |
| 경상북도 | 2,551,370 | 1,288,999 | 1,262,371 | | | | |
| 경상남도 | 3,248,703 | 1,636,010 | 1,612,693 | | | | |
| 4) 제주특별자치도 | 674,353 | 337,578 | 336,775 | | | | |

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 2. 검색 및 URL 생성



https://kosis.kr/openapi/Param/statisticsParameterData.do?method=getList&apiKey=MjUzMTdiMjcyZjQwY2VkMWJkY2RkYTRmZDkwMWQ3ZGY=&itmId=T20+T21+&objL1=00+11+26+27+28+29+30+31+36+41+51+43+44+52+46+47+48+50+&objL2=&objL3=&objL4=&objL5=&objL6=&objL7=&objL8=&format=json&jsonVD=Y&prdSe=M&newEstPrdCnt=3&orgId=101&tblId=DT_1B040A3

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 3. JSON 데이터 이해

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

[{"TBL_NM":"행정구역(시군구)별 성별 인구수","PRD_DE":"202202","TBL_ID":"DT_1B040A3","ITM_NM":"총인구수","ITM_NM_ENG":"Koreans (Total)","ITM_ID":"T20","UNIT_NM":"명","ORG_ID":"101","UNIT_NM_ENG":"Person","C1_OBJ_NM":"행정구역(시군구)별","C1_OBJ_NM_ENG":"By Administrative District","DT":"51625561","PRD_SE":"M","C1":"00","C1_NM":"전국","C1_NM_ENG":"Whole country"},{"TBL_NM":"행정구역(시군구)별 성별 인구수","PRD_DE":"202202","TBL_ID":"DT_1B040A3","ITM_NM":"남자인구수","ITM_NM_ENG":"Koreans (Male)","ITM_ID":"T21","UNIT_NM":"명","ORG_ID":"101","UNIT_NM_ENG":"Person","C1_OBJ_NM":"행정구역(시군구)별","C1_OBJ_NM_ENG":"By Administrative District","DT":"25738197","PRD_SE":"M","C1":"00","C1_NM":"전국","C1_NM_ENG":"Whole country"},{"TBL_NM":"행정구역(시군구)별 성별 인구수","PRD_DE":"202202","TBL_ID":"DT_1B040A3","ITM_NM":"여자인구수","ITM_NM_ENG":"Koreans (Female)","ITM_ID":"T22","UNIT_NM":"명","ORG_ID":"101","UNIT_NM_ENG":"Person","C1_OBJ_NM":"행정구역(시군구)별","C1_OBJ_NM_ENG":"By Administrative District","DT":"25887364","PRD_SE":"M","C1":"00","C1_NM":"전국","C1_NM_ENG":"Whole country"},{"TBL_NM":"행정구역(시군구)별 성별 인구

| TBL_NM | PRD_DE |
|---|---|
| ABC1 | 202201 |
| ABC2 | 202202 |
| ABC3 | 202203 |

Object 형식
Key/value 형태의 구조

```
{
    key1: value,
    key2: value
}
```

[ {TBL_NM": "ABC1", "PRD_DE" : "202201"}, 1행 데이터
{TBL_NM": "ABC2", "PRD_DE" : "202202"}, 2행 데이터
{TBL_NM": "ABC3", "PRD_DE" : "202203"} ] 3행 데이터

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 3. JSON 데이터 이해

http://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=MjUzMTdiMjcyZjQwY2VkMWJkY2RkYTRmZDkwMWQ3ZGY=&format=jsona

kosis.kr

1행

[{"TBL_NM":"행정구역(시군구)별 성별 인구수","PRD_DE":"202002","TBL_ID":"DT_1B040A3","ITM_NM":"총인구수","ITM_NM_ENG":"Koreans (Total)","ITM_ID":"T20","UNIT_NM":"명","ORG_ID":"101","UNIT_NM_ENG":"Person","C1_OBJ_NM":"행정구역(시군구)별","C1_OBJ_NM_ENG":"By Administrative District","DT":"51844627","PRD_SE":"M","C1":"00","C1_NM":"전국","C1_NM_ENG":"Whole country"},

2행

{"TBL_NM":"행정구역(시군구)별 성별 인구수","PRD_DE":"202002","TBL_ID":"DT_1B040A3","ITM_NM":"총인구수","ITM_NM_ENG":"Koreans (Total)","ITM_ID":"T20","UNIT_NM":"명","ORG_ID":"101","UNIT_NM_ENG":"Person","C1_OBJ_NM":"행정구역(시군구)별","C1_OBJ_NM_ENG":"By Administrative District","DT":"9736962","PRD_SE":"M","C1":"11","C1_NM":"서울특별시","C1_NM_ENG":"Seoul"},

# 2. 국가통계포털 API 데이터 불러오기 (JSON)

## 4. Python 활용 불러오기

```python
# 라이브러리 선언
import pandas as pd

# 데이터 접속 주소 입력
url = "요청주소"
json = pd.DataFrame()
try:
    json = pd.read_json(url)
except Exception as e:
    print(e)
json
```

json 형태는 바로 변환 가능

```python
# 라이브러리 선언
import pandas as pd
import requests

# 데이터 접속 주소 입력
url = "http://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=MjUzMTdiMjcyZjQwY2VkMWJkY2RkY

json = pd.DataFrame()

try:
    response = requests.get(url)
    if response.status_code == 200:
        json = pd.read_json(url)
    else:
        response.close()
except Exception as e:
    print(e)
json
```

| | TBL_NM | PRD_DE | TBL_ID | ITM_NM | ITM_NM_ENG | ITM_ID | UNIT_NM | ORG_ID | UNIT_NM_ENG | C1_OBJ_NM |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 행정구역(시군구)별 성별 인구수 | 202003 | DT_1B040A3 | 총인구수 | Koreans (Total) | T20 | 명 | 101 | Person | 행정구역(시군구)별 |

www.youtube.com/hkcode

# 실습

## 검색: "소비자물가지수"

**"소비자물가지수"**에 대한 검색결과는 **683건**입니다.

**통계표 : 130건**

- 소비자물가지수(2020=100)
- 소비자물가지수(시도)
- 자가주거비포함 소비자물가지수(2020=100)
- 지출목적별 소비자물가지수(품목포함, 2020=100)
- 품목별 소비자물가지수(품목성질별: 2020=100)
- 품목성질별 소비자물가지수(2020=100)

| | TBL_NM | PRD_DE | TBL_ID | ITM_NM | ITM_NM_ENG | ITM_ID | UNIT_NM | ORG_ID | UNIT_NM_ENG | C1_OBJ_NM | C1_OBJ_NM_ENG | DT | PRD_SE | C1 | C1_NM | C1_NM_ENG | LST_CHN_DE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 소비자물가지수(시도) | 202401 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 113.15 | M | T10 | 전국 | whole country | 2024-01-31 |
| 1 | 소비자물가지수(시도) | 202402 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 113.77 | M | T10 | 전국 | whole country | 2024-03-04 |
| 2 | 소비자물가지수(시도) | 202403 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 113.94 | M | T10 | 전국 | whole country | 2024-03-29 |
| 3 | 소비자물가지수(시도) | 202401 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 112.59 | M | T11 | 서울특별시 | Seoul | 2024-01-31 |
| 4 | 소비자물가지수(시도) | 202402 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 113.05 | M | T11 | 서울특별시 | Seoul | 2024-03-04 |
| 5 | 소비자물가지수(시도) | 202403 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 113.00 | M | T11 | 서울특별시 | Seoul | 2024-03-29 |
| 6 | 소비자물가지수(시도) | 202401 | INH_1J22003 | 소비자물가지수(총지수) | CPI | T | 2020=100 | 101 | 2020=100 | 시도별 | by City | 113.10 | M | T12 | 부산광역시 | Busan | 2024-01-31 |

www.youtube.com/nkcode

# 실습

## 검색: "소비자물가지수"

https://kosis.kr/openapi/Param/statisticsParameterData.do?method=getList&apiKey=MjUzMTdiMjcyZjQwY2VkMWJkY2RkYTRmZDkwMWQ3ZGY=&itmId=T+&objL1=ALL&objL2=&objL3=&objL4=&objL5=&objL6=&objL7=&objL8=&format=json&jsonVD=Y&prdSe=M&newEstPrdCnt=3&orgId=101&tblId=INH_1J22003

# 파트3. 공공데이터 포털 API 데이터 불러오기

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 모듈 개요

[과정개요]

공공데이터 포털 API 데이터 불러오기(XML)

[교육목표]

- 웹 상의 공개된 API 활용방법

[교육대상]

- 데이터 분석가 / 인공지능 전문가
- 데이터 엔지니어

| 내용 | 학습내용 |
|---|---|
| **공공데이터 포털 API 데이터 불러오기** | - API로 공개한 데이터의 수집방법을 실습합니다. (XML 타입) |

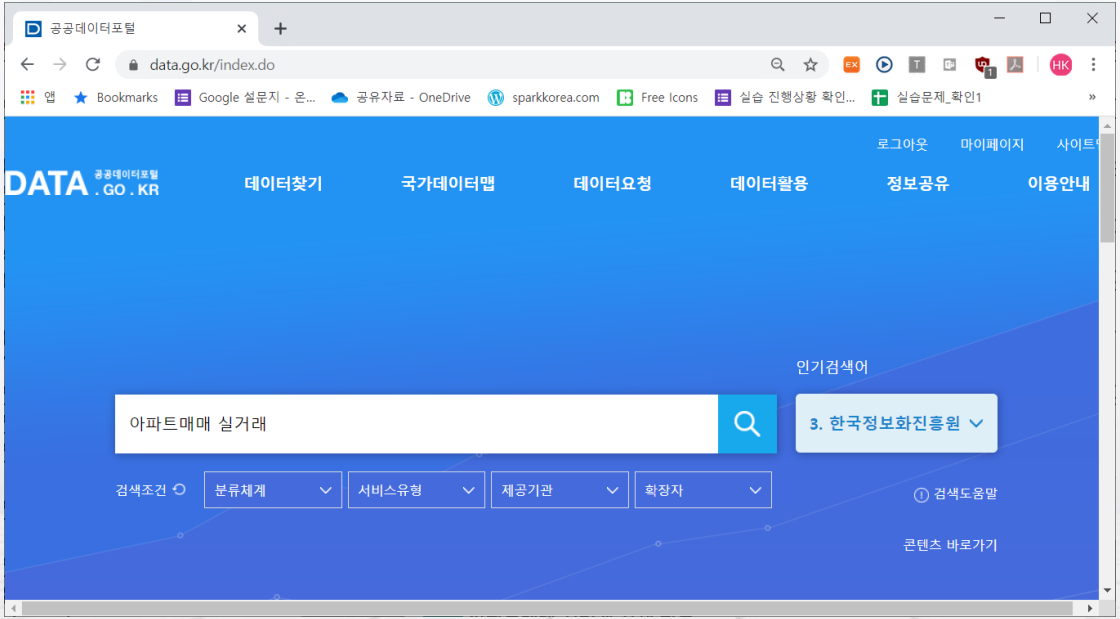
# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## XML 데이터 불러오는 순서

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

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

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

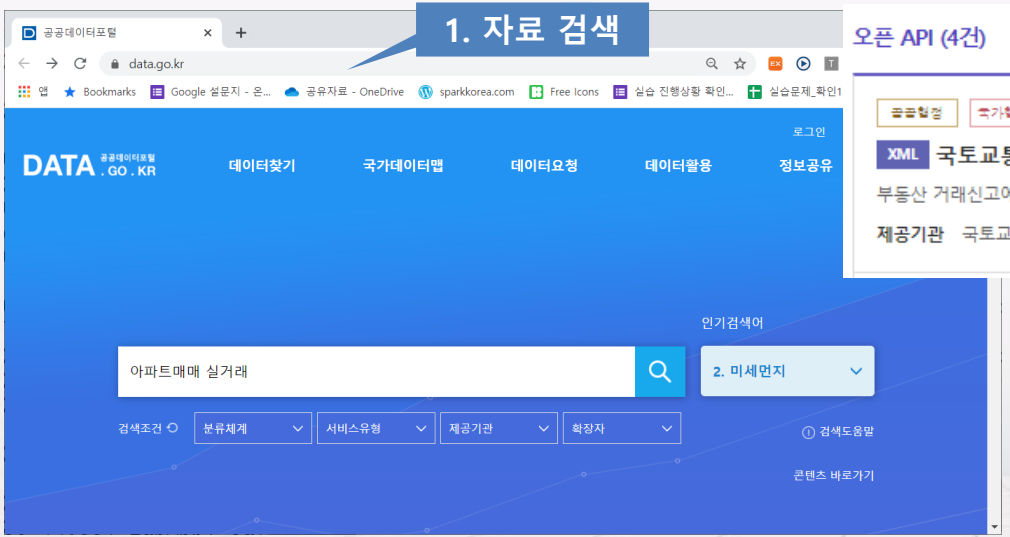


# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

- 작업시간 : `24.4.19(금) 22:00 ~ `24.4.20(토) 6:00 (8시간)
- 대상 API : [국토교통부] 실거래 관련 API 13 종
  1. 국토교통부_연립다세대 전월세 자료
  2. 국토교통부_아파트 전월세 자료
  3. 국토교통부_단독/다가구 전월세 자료
  4. 국토교통부_오피스텔 전월세 신고 조회 서비스
  5. 국토교통부_토지 매매 신고 조회 서비스
  6. 국토교통부 _연립다세대 매매 실거래자료
  7. 국토교통부_상업업무용 부동산 매매 신고 자료
  8. 국토교통부_단독/다가구 매매 실거래 자료
  9. 국토교통부_오피스텔 매매 신고 조회 서비스
  10. 국토교통부_아파트매매 실거래자료
  11. 국토부_아파트매매 실거래 상세 자료
  12. 국토교통부_아파트 분양권전매 신고 자료
  13. 국토교통부_공장 및 창고 등 부동산 매매 신고 자료

## 1. 자료검색 및 제공유형 확인

### 검색어로 데이터 탐색 (아파트매매 실거래)



http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev?LAWD_CD=11110&DEAL_YMD=202403&serviceKey=AT33s775KYpJOkUBJu0dxkJuUeIfDIOJRzAH084EQS3JN%2BzFjErLHuk%2FGZa9L4gBTSGCzeA69tI9PwLp7B37IQ%3D%3D

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 2. 데이터 활용신청

### 활용신청 시 로그인 요청

상세기능 | 활용사례 | 추천데이터

**상세기능**

목록 국토교통부_아파트매매 상세자료 조회

**지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료를 제공하는 아파트 매매 신고 정보 조회**

- **활용승인 절차** 개발단계 : 허용 / 운영단계 : 허용
- **신청가능 트래픽** 1,000 / 운영계정은 활용사례 등록시 신청하면 트래픽 증가 가능
- **요청주소**
- **서비스URL** http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev

활용신청

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 2. 데이터 활용신청

### 시스템유형, 정보, 라인센스 동의 후 신청

**OpenAPI 개발계정 신청**

XML 아파트매매 실거래자료

| | | | |
|---|---|---|---|
| 제공기관 | 국토해양부 | 서비스유형 | REST |
| 심의여부 | 자동승인 | 신청유형 | 개발계정 \| 활용신청 |
| 처리상태 | 신청 | 활용기간 | 승인일로부터 24개월 간 활용가능 |

**공공데이터 제공제도**

* 공공데이터중 위치정보를 포함한 서비스를 사용하고자 하는 사업자는 '위치정보의 보호 및 이용 등에 관한 법률'에 따라 방송통신위원회에 **비스 허가**'를 받거나 '**위치기반 서비스사업 신고**'를 하여야 합니다.
* 이에 해당하는 사업자인 경우에는 첨부파일에 '**위치기반서비스사업신고필**
* 활용신청 시 '**위치기반서비스사업신고필증**'이 등록되지 않으면 반려가 될

**활용목적 선택**

*표시는 필수 입력

*활용목적 ○ 웹 사이트 개발 ○ 앱개발 (모바일,솔루션등) ○ 기타 ○ 참고자료 ◉ 연구(논문 등)

교육용 학습 데이터 활용

**활용목적: 연구**
**교육용 학습 데이터 활용**

**시스템유형 선택**

시스템 유형 ◉ 일반 ○ 서버구축

**1. 시스템 유형: 일반**

[안내] 서버 구축 여부에 따라 시스템 유형을 입력하실 수 있습니다.
- 일반 : OPEN API 서비스를 호출하여 응답 받은 결과값을 서버에 저장하지 않고 사용할 겨우 (서버 미구축)
- 서버구축 : OPEN API 서비스를 호출하여 응답 받은 결과값을 서버에 저장하거나 DB화하여 사용할 경우

**상세기능정보 선택**

| ☑ | 상세기능 | 설명 | 일일 트래픽 |
|---|---|---|---|
| ☑ | 아파트 매매 자료 | 지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료를 제공하는 아파트 매매 신고 정보 조회 | 1000 |

**라이선스 표시**

*이용허락범위 이용허락범위 제한 없음
☑ 동의합니다.

**2. 체크 및 신청**

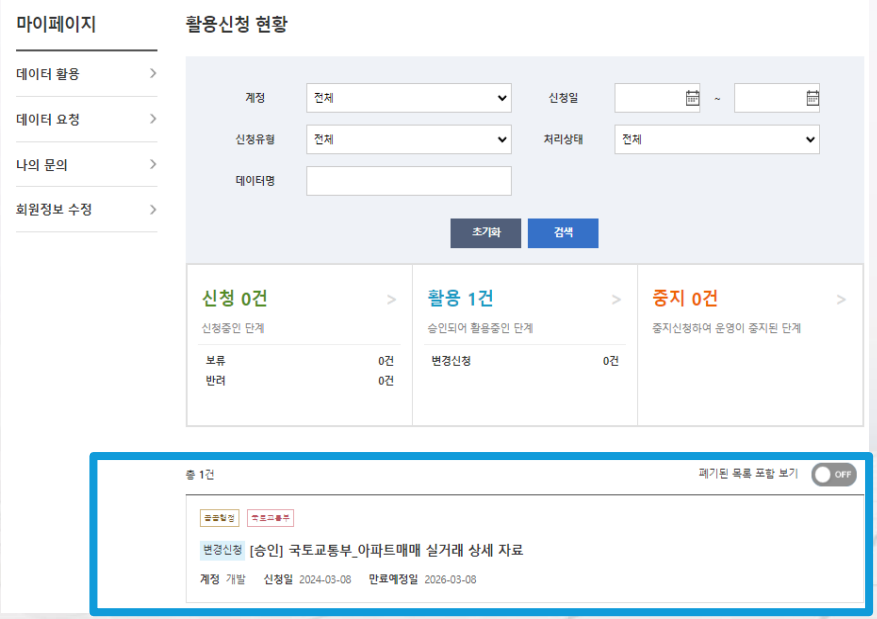
취소 활용신청

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

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 3. 개발계정 API 키 획득

마이페이지 → 데이터활용 → Open API → 활용 신청현황 신청자료 선택 → 일반 인증키 받기



서비스정보

| | |
|---|---|
| 참고문서 | 아파트 매매 상세자료 조회 기술문서.hwp |
| 데이터포맷 | XML |
| End Point | http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev?_wadl&type=xml |

API 환경 또는 API 호출 조건에 따라 인증키가 적용되는 방식이 다를 수 있습니다.
포털에서 제공되는 Encoding/Decoding 된 인증키를 적용하면서 구동되는 키를 사용하시기 바랍니다.
* 향후 포털에서 더 명확한 정보를 제공하기 위해 노력하겠습니다.

| | |
|---|---|
| 일반 인증키 (Encoding) | sjOpJdbWHR3lhq4f8w5Dg6lc6IDZPdXeue%2BzGqY47h4Jp4sKj8SE5wKtpoHT7yfpI1NMFLyMMYHSiY4MNP6dhQ%3D%3D |
| 일반 인증키 (Decoding) | sjOpJdbWHR3lhq4f8w5Dg6lc6IDZPdXeue+zGqY47h4Jp4sKj8SE5wKtpoHT7yfpI1NMFLyMMYHSiY4MNP6dhQ== |

활용신청 상세기능정보

| NO | 상세기능 | 설명 | 일일 트래픽 | 미리보기 |
|---|---|---|---|---|
| 1 | 국토교통부_아파트매매 상세자료 | 지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료를 제공하는 아파트 매매 신고 정보 조회 | 1000 | 확인 |

요청변수(Request Parameter) 닫기

| 항목명 | 샘플데이터 | 설명 |
|---|---|---|
| ServiceKey | sjOpJdbWHR3lhq4f8w5D | 공공데이터포털에서 받은 인증키 |
| pageNo | 1 | 페이지번호 |
| numOfRows | 10 | 한 페이지 결과 수 |
| LAWD_CD | 11110 | 지역코드 |
| DEAL_YMD | 201512 | 계약월 |

미리보기

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 3. 개발계정 API 키 획득

마이페이지 → 오픈API → 신청자료 선택 –〉 일반 인증키 받기

### 서비스정보

| | |
|---|---|
| 일반 인증키 (UTF-8) | MYHSiY4MNP6dhQ%3D%3D 복사 |
| End Point | http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev?_wadl&type=xml |
| 데이터포맷 | XML |
| 참고문서 | 아파트 매매 상세자료 조회 기술문서.hwp |

### 상세기능정보

개발가이드

| NO | 상세기능 | 설명 | 활용제한 여부 | 일일 트래픽 | 심의결과 | 미리보기 다운로드 |
|---|---|---|---|---|---|---|
| 1 | 아파트매매 상세자료 | 지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료를 제공하는 아파트 매매 신고 정보 조회 | - | 1000 | 승인 | 실행 |

### 아파트매매 상세자료

지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료를 제공하는 아파트 매매 신고 정보 조회

- **활용승인 절차** 개발계정 : 자동승인 / 운영계정 : 자동승인
- **신청가능 트래픽** 1000000 / 운영계정은 활용사례 등록시 신청하면 트래픽 증가 가능
- **요청주소** http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev
- **서비스URL**

활용신청 뒤로가기

### 요청변수(Request Parameter)

| 항목명(국문) | 항목명(영문) | 항목크기 | 항목구분 | 샘플데이터 | 항목설명 |
|---|---|---|---|---|---|
| 서비스키 | ServiceKey | 20 | 필수 | - | 공공데이터포털에서 받은 인증키 |
| 페이지 번호 | pageNo | 4 | 옵션 | 1 | 페이지번호 |
| 한 페이지 결과 수 | numOfRows | 4 | 옵션 | 10 | 한 페이지 결과 수 |
| 지역코드 | LAWD_CD | 5 | 필수 | 11110 | 지역코드 |
| 계약월 | DEAL_YMD | 6 | 필수 | 201512 | 계약월 |

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 4. 개발문서 및 개발가이드 확인

마이페이지 → 오픈API → 개발계정 상세보기

http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev?LAWD_CD=11110&DEAL_YMD=202403&serviceKey=AT33s775KYpJOkUBJu0dxkJuUelfDIOJRzAH084EQS3JN%2BzFjErLHuk%2FGZa9L4gBTSGCzeA69tI9PwLp7B37IQ%3D%3D

**기본정보**

| | | | | | |
|---|---|---|---|---|---|
| 서비스명 | 아파트매매 실거래 상세 자료 상세설명 | | | | |
| 서비스 유형 | REST | 일일트래픽 | 0 | 평균응답속도(초) | 0 |
| 심의여부 | 자동승인 | 신청유형 | 개발계정 \| 활용신청 | 처리상태 | 승인 |
| 활용기간 | 2020-04-09 ~ 2022-04-09 | | | | |

**서비스정보**

| | |
|---|---|
| 일반 인증키 (UTF-8) | |
| End Point | http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev?_wadl&type=xml |
| 데이터포맷 | XML |
| 참고문서 | 아파트 매매 상세자료 조회 기술문서.hwp |

1. 참고문서

**상세기능정보** 개발가이드

| NO | 상세기능 | 설명 | 활용제한 여부 | 일일 트래픽 | 심의결과 | 미리보기 다운로드 |
|---|---|---|---|---|---|---|
| 1 | 아파트매매 상세자료 | 지역코드와 기간을 이용하여 해당기간, 해당지역의 아파트 매매 신고자료를 제공하는 아파트 매매 신고 정보 조회 | - | 1000 | 승인 | 실행 |

ㅇ 응답메시지 예제

| REST(URI) |
|---|
| http://openapi.molit.go.kr:8081/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTrade?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)

## 4. 개발문서 및 개발가이드 확인

나. 상세기능 목록

| 번호 | API명(국문) | 상세기능명(영문) | 상세기능명(국문) |
|---|---|---|---|
| 1 | 아파트 매매 신고 상세자료 | getRTMSDataSvcAptTradeDev | 아파트 매매 신고 상세자료 |

다. 상세기능내역

a) 상세기능정보

| 상세기능 번호 | 1 | 상세기능 유형 | 조회 (자료) |
|---|---|---|---|
| 상세기능명(국문) | 아파트 매매 신고 상세자료 조회 | | |
| 상세기능 설명 | 행정표준코드관리시스템(www.code.go.kr)의 법정동 코드 중 앞5자리(예시 : 서울 종로구 - 11110), 계약년월(예시 : 201801)로 해당 지역, 해당 기간의 아파트 매매신고 상세자료 조회 | | |
| Call Back URL | http://openapi.molit.go.kr/OpenAPI_ToolInstallPackage/service/rest/RTMSOBJSvc/getRTMSDataSvcAptTradeDev | | |

b) 요청 메시지 명세

| 항목명(영문) | 항목명(국문) | 항목 크기 | 항목 구분* | 샘플 데이터 | 항목설명 |
|---|---|---|---|---|---|
| serviceKey | 인증키 | 100 | 1 | 인증키 (URL Encode) | 공공데이터포털에서 발급받은 인증키 |
| pageNo | 페이지번호 | 4 | 0 | 1 | 페이지번호 |
| numOfRows | 한 페이지 결과 수 | 4 | 0 | 10 | 한 페이지 결과 수 |
| LAWD_CD | 지역코드 | 5 | 1 | 11110 | 각 지역별 코드 행정표준코드관리시스템(www.code.go.kr)의 법정동코드 10자리 중 앞 5자리 |
| DEAL_YMD | 계약월 | 6 | 1 | 201512 | 실거래 자료의 계약년월(6자리) |

※ 항목구분 : 필수(1), 옵션(0), 1건 이상 복수건(1..n), 0건 또는 복수건(0..n)

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 4. 개발문서 및 개발가이드 확인

서비스 URL 주소

http://openapi.molit.go.kr/..TradeDev

파라미터

| 항목명 | 구분 | 샘플 | 설명 |
|---|---|---|---|
| ServiceKey | 필수 | ABC | 인증키 |
| LAWD_CD | 필수 | 11110 | 지역코드 |
| DEAL_YMD | 필수 | 201512 | 계약월 |

http://openapi.molit.go.kr/..TradeDev?ServiceKey=ABC&LAWD_CD=11110&DEAL_YMD=201512

주소와 파라미터 구분 "?"

파라미터 구분 "&"

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 5. XML 태그 이해

Extensible Markup Language로 다목적 마크업 언어이다.
(마크업 언어는 태그 등을 이용하여 데이터의 구조를 기술하는 언어 임)


```
▼<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>
         <도로명건물본번호코드>00004</도로명건물본번호코드>
         <도로명건물부번호코드>00000</도로명건물부번호코드>
         <도로명시군구코드>11110</도로명시군구코드>
         <도로명일련번호코드>03</도로명일련번호코드>
         <도로명지상지하코드>0</도로명지상지하코드>
         <도로명코드>4100135</도로명코드>
         <법정동> 사직동</법정동>
         <법정동본번코드>0009</법정동본번코드>
         <법정동부번코드>0000</법정동부번코드>
         <법정동시군구코드>11110</법정동시군구코드>
         <법정동읍면동코드>11500</법정동읍면동코드>
         <법정동지번코드>1</법정동지번코드>
         <아파트>광화문풍림스페이스본(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>
      ▼<item>
         <거래금액> 60,000</거래금액>
         <건축년도>1981</건축년도>
```


body → items → item 반복

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (전략)


```
▼<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>
```


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

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (전략)

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

전체 행(row) 담을 리스트

각 행(row)별 컬럼 값 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)

## 6. 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+queryParams)
response.encoding='utf-8'
xmlobj = BeautifulSoup(resp.text,"lxml-xml")
xmlobj
```

패키지 인스톨 필요!
pip install bs4
**pip install lxml --force –U 설치 후 kernel restart**


'<?xml version="1.0" encoding="UTF-8" standalone="yes"?><response><header><resultCode>00</resultCode><resultMsg>NORMAL SE
</header><body><items><item><거래금액>  82,500</거래금액><건축년도>2008</건축년도><년>2015</년><도로명>사직로8길</도로
코드>00004</도로명건물본번호코드><도로명건물부번호코드>00000</도로명건물부번호코드><도로명시군구코드>11110</도로명시군구
코드>03</도로명일련번호코드><도로명지상지하코드>0</도로명지상지하코드><도로명코드>4100135</도로명코드><법정동> 사직동</법
드>0009</법정동본번코드><법정동부번코드>0000</법정동부번코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드
코드><법정동지번코드>1</법정동지번코드><아파트>광화문풍림스페이스본(101동~105동)</아파트><월>12</월><일>1~10</일><일련번
번호><전용면적>94.51</전용면적><지번>9</지번><지역코드>11110</지역코드><층>11</층></item><item><거래금액>  60,000</거래
</건축년도><년>2015</년><도로명>세종대로23길</도로명><도로명건물본번호코드>00047</도로명건물본번호코드><도로명건물부번호
물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일련번호코드>02</도로명일련번호코드><도로명지상지하코드>0</
<도로명코드>4100190</도로명코드><법정동> 당주동</법정동><법정동본번코드>0145</법정동본번코드><법정동부번코드>0000</법정동
군구코드>11110</법정동시군구코드><법정동읍면동코드>11700</법정동읍면동코드><법정동지번코드>1</법정동지번코드><아파트>롯데
아파트><월>12</월><일>21~31</일><일련번호>11110-12</일련번호><전용면적>149.95</전용면적><지번>145</지번><지역코드>11110</
</item><item><거래금액>  130,000</거래금액><건축년도>2004</건축년도><년>2015</년><도로명>경희궁2길</도로명><도로명건물본
로명건물본번호코드><도로명건물부번호코드>00005</도로명건물부번호코드><도로명시군구코드>11110</도로명시군구코드><도로명일
명일련번호코드><도로명지상지하코드>0</도로명지상지하코드><도로명코드>4100005</도로명코드><법정동> 내수동</법정동><법정동
동본번코드><법정동부번코드>0015</법정동부번코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드>11800</법정동


```
C:\Users\kopo>pip show lxml
Name: lxml
Version: 4.6.1
Summary: Powerful and Pythonic XML processing library combining
libxml2/libxslt with Etree and lxml.obje
Home-page: https://lxml.de/
Author: lxml dev team
Author-email: lxml-dev@lxml.de
License: BSD
Location: c:\Users\kopo\anaconda3\lib\site-packages
Requires:
Required-by: object-detection

C:\Users\kopo>_
```

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (Part #1)

```
#Used to service API connection
import requests
from lxml import html
from bs4 import BeautifulSoup
import pandas as pd
from urllib.request import Request, urlopen
from urllib.parse import urlencode, quote_plus, unquote

API_Key = unquote('key')
url = "serviceUrl"
queryParams = '?' + urlencode(
    {
        quote_plus('ServiceKey') : API_Key,
        quote_plus('LAWD_CD'): '11110',
        quote_plus('DEAL_YMD'): '202403'
    }
)
response = requests.get(url+queryParams)
response.encoding='utf-8'
xmlobj = BeautifulSoup(response.text,"lxml-xml")
```

패키지 인스톨 필요!
pip install bs4
**pip install lxml --force –U 설치 후 kernel restart**

**Url인코딩**

**Get 방식을 통해 HTTP 요청 시 파라미터가 붙는경우가 생기는데 URL은 ASCII 코드값만 사용된다!**
**(*아스키코드: 미국 정보교환용 7비트 부호)**

**쿼리파라미터에 한글이 포함될경우 ASCII 코드만드로 표현할수 없어서 인코딩이 진행됨!**

**호출하는 API 마다 알아서 지원되는 경우도 있지만 인코딩 형태로 전송하는 것이 좋다!**

**URL 인코딩은 URL에서 사용할 수 없는 문자들을 안전하게 전송하기 위해 문자열을 변환하는 프로세스입니다. 예를 들어, URL에서 특별한 의미를 갖는 문자인 '='나 '&'과 같은 문자는 인코딩되어야 합니다**

www.youtube.com/hkcode

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (Part #2)

```
# 전체 Row를 담을 리스트
rowList = []

# 한 행의 컬럼값을 담을 리스트
columnList = []

# 행 값을 담고 있는 모든 태그를 검색
rows = xmlobj.findAll('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><법정동본번코드>0009</법정동본번코드><법정동부번코드>0000</법정동부번코드><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>94.5</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><법정동본번코드>0145</법정동본번코드><법정동부번코드>0000</법정동부번코드><법정동시군구코드>11110</법정동시군구코드><법정동읍면동코드>11700</법정동읍면동코드><법정동지번코드>1</법정동지번코드><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)

## 6. Python 활용 불러오기 (Part #3)

**한개 행의 모든 컬럼값을 컬럼 리스트에 담는다!**

```
# 한개 행의 모든컬럼 값을 리스트에 담아보자!
columns = rows[0].findAll()

# 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)
```

태그 내 text -> .text
태그 내 태그명 -> .name

```
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>,
 <법정동본번코드>0009</법정동본번코드>,
 <법정동부번코드>0000</법정동부번코드>,
 <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>]
```

```
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)

## 6. Python 활용 불러오기 (Part #4)

모든 행을 반복해서 담는다 단, 다음 행 시작 시 컬럼리스트를 비운다.

```
# 전체 Row를 담을 리스트
rowList = []
# 한 행의 컬럼값을 담을 리스트
columnList = []
# 행 값을 담고 있는 모든 태그를 검색
rows = xmlobj.findAll('item')

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

문자열 공백제거 replace 또는 strip()

```
1  dataframeList
```

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

각, 김효관 교수)
www.youtube.com/hkcode

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (Part #5)

```
# 데이터 프레임으로 변환하자
pd.DataFrame(rowList)
```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 82,500 | 2008 | 2015 | 사직로8길 | 00004 | 00000 | 11110 | 03 | 0 | 4100135 | ... | 11500 | 1 |

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (리펙토링)

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

rows를 한개씩 꺼내서
eachrow변수에 담는다.

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

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (컬럼이름 추출)

```
# 전체 Row를 담을 리스트
rowList = []

# 한 행의 컬럼값을 담을 리스트
columnList = []

# 행 값을 담고 있는 모든 태그를 검색
rows = xmlobj.findAll('item')

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

```
for eachrow in rows:
    columns = eachrow.findAll()
    for eachColumns in columns:
        if(rowcnt==0):
            columnNameList.append(eachColumns.name)
        eachcolumnValue = eachColumns.text
        columnList.append(eachcolumnValue)
    rowcnt=rowcnt+1
    rowList.append(columnList)
    columnList = []
# 데이터 프레임으로 변환하자
result2 = pd.DataFrame(rowList, columns = columnNameList)
result2.head()
```

# 3. 공공데이터 포털 API 데이터 불러오기 (XML)

## 6. Python 활용 불러오기 (컬럼이름 맵핑)

```
1 result2
```

| | 거래금액 | 건축년도 | 년 | 도로명 | 도로명건물본번호코드 | 도로명건물부번호코드 | 도로명시군구코드 | 도로명일련번호코드 | 도로명지상지하코드 | 도로명코드 | ... | 법정동읍면동코드 | 법정동지번코드 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 82,500 | 2008 | 2015 | 사직로8길 | 00004 | 00000 | 11110 | 03 | 0 | 4100135 | ... | 11500 | 1 |
| **1** | 60,000 | 1981 | 2015 | 세종대로23길 | 00047 | 00000 | 11110 | 02 | 0 | 4100190 | ... | 11700 | 1 |

# 파트4. XML 데이터 수집 응용 실습

# 실습

## "약국" 검색 후

오픈 API (43건)

보건의료 공공기관

XML 국립중앙의료원_전국 약국 정보 조회 서비스

전국 약국 정보를 조회하기 위한 서비스로서 주소별, 진료요일별 및 시간대로 조회할 수 있다

**제공기관** 국립중앙의료원 **수정일** 2023-12-13 **조회수** 34753 **활용신청** 5219 **키워드** 약국,약국정보,약국조회

(가) 요청 메시지 명세

| 항목명(영문) | 항목명(국문) | 항목크기 | 항목구분 | 샘플데이터 | 항목설명 |
|---|---|---|---|---|---|
| Q0 | 주소(시도) | 16 | 0 | 서울특별시 | 주소(시도) |
| Q1 | 주소(시군구) | 60 | 0 | 강남구 | 주소(시군구) |
| QT | 진료요일 | 1 | 0 | 1 | 월~일요일, 공휴일: 1~8 |
| QN | 기관명 | 100 | 0 | 삼성약국 | 기관명 |
| ORD | 순서 | 20 | 0 | NAME | 순서 |
| pageNo | 페이지 번호 | 100 | 0 | 1 | 페이지 번호 |
| numOfRows | 목록 건수 | 100 | 0 | 10 | 목록 건수 |

※ 항목구분 : 필수(1), 옵션(0), 1건 이상 복수건(1..n), 0건 또는 복수건(0..n)

**http://apis.data.go.kr/B552657/ErmctInsttInfoInqireService/getParmacyListInfoInqire?ServiceKey=AT33s775KYpJOkUBJu0dxkJuUeIfDIOJRzAH084EQS3JN%2BzFjErLHuk%2FGZa9L4gBTSGCzeA69tI9PwLp7B37IQ%3D%3D&WGS84_LON=127.085156592737%26WGS84_LAT%3D37.4881325624879&WGS84_LAT=37.4881325624879**

# 실습

```
pd.DataFrame(rowList, columns = nameList)
```

| | dutyaddr | dutyetc | dutyname | dutytel1 | dutytime1c | dutytime1s | dutytime2c | dutytime2s | dutytime3c | dutytime3s | ... | dutytime7c | dutytime7s |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 충청남도 서산시 지곡면 충의로 1192 | 전화문의 필요 | (새)지곡백화점약국 | 041-662-2808 | 2200 | 0800 | 2200 | 0800 | 2200 | 0800 | ... | 2200 | 0800 |
| **1** | 경기도 안양시 만안구 박달로 518 (박달동) | 1(일)약국 | 031-468-3354 | 2000 | 0900 | 2000 | 0900 | 2000 | 0900 | 2000 | ... | 140 | 09 |

# 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

# 감사합니다