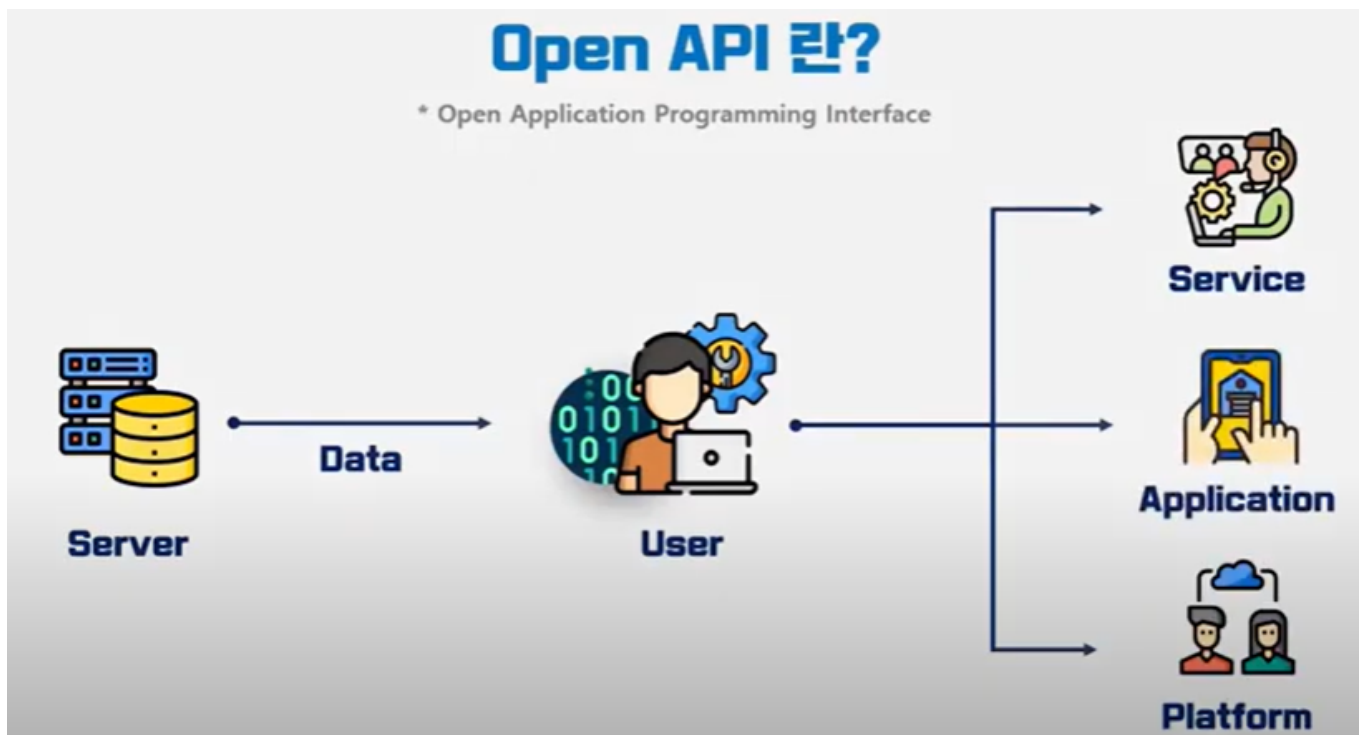


오픈 API를 이용한 웹페이지 크롤링

오픈 API란?

- 크롤링은 하면 웹사이트에 과부하가 생기므로 이를 방지하는 차원에서 크롤링을 막는 경우가 있음
- 오픈 API를 사용하여 원하는 정보를 얻을 수 있게 도와 줌
- API : Application Programming Interface
 - 사전적정의 : 운영체제나 프로그래밍 언어가 제공하는 기능을 응용프로그램에서 사용 또는 제어할 수 있도록 만든 인터페이스



*출처: 공공데이터활용지원센터, <https://www.youtube.com/watch?v=pe5-RQb-OPI>
(<https://www.youtube.com/watch?v=pe5-RQb-OPI>)

공공데이터 포털 사이트(<http://data.go.kr> (<http://data.go.kr>))

국가중점데이터 분류 조건 추가하기 +

조건열기 ▾

전체(73,777건)	파일데이터(55,601건)	오픈 API(9,792건)	표준데이터셋169개(8,384건)
			수정일자순 ▾ 10개씩 ▾ 정렬

오픈 API (9,792건)

공공행정	국가행정기관	미리보기
XML	행정안전부_통계연보_주민등록기준 연령별 1인 세대수	
주민등록세대 기준 10대 이하, 20대, 30대, 40대 등 나이를 기준으로 연령별 1인세대 정보를 포함하는 통계 정보를 제공하는 서비스		
제공기관	행정안전부	수정일 2022-11-21 조회수 154 활용신청 6 키워드 연령,1인세대,주민과
활용신청		
공공행정	국가행정기관	미리보기
XML	행정안전부_통계연보_지역별 새마을금고 운영 지점수	
서울, 부산, 대구, 인천, 충북, 제주, 광주·전남 등 지역별 새마을금고 운영 내용을 포함하는 통계 정보를 제공하는 서비스		
제공기관	행정안전부	수정일 2022-11-21 조회수 30 활용신청 1 키워드 새마을금고,지역,지역금융지원과
활용신청		
공공행정	국가행정기관	미리보기
XML	행정안전부_통계연보_지역별 정보화마을 개수	
서울, 부산, 대구, 인천, 대전, 세종, 충북, 충남 등 지역을 기준으로 정보화마을 정보를 포함하는 통계 정보를 제공하는 서비스		
제공기관	행정안전부	수정일 2022-11-21 조회수 21 활용신청 1 키워드 지역,지역공동체과,정보화
의견수렴 게시판		

공공데이터 포털의 오픈API 사용과정



실습. 부동산 데이터 웹크롤링

<https://www.data.go.kr/data/15099361/openapi.do> (<https://www.data.go.kr/data/15099361/openapi.do>)

1. 공공데이터 포털사이트에서 API 등록하기

1. 회원가입 : 오픈 API 사용을 위해 회원가입
2. 로그인
3. data.go.kr 검색란에서 '부동산 거래 통계 조회 서비스' 입력
4. 검색 결과 중 [오픈API]에서 '한국부동산원_부동산 거래 통계 조회 서비스' 선택한 후 <활용신청>버튼을 클릭
5. 활용신청 페이지에서 활용목적을 입력하고 '동의합니다'체크 후 <활용신청>버튼 클릭

6. [마이페이지]로 이동하여 **[오픈API]-[인증키 발급현황]** 메뉴 클릭

- API를 사용하기 위해 필요한 인증키임

7. [마이페이지]로 이동하여 승인된 목록에서 **[부동산 거래현황 통계조회서비스]** 를 클릭

- 서비스 정보 확인

8. **[기본정보]** 의 **<상세설명>** 버튼을 클릭하여 '부동산 거래현황 통계조회서비스' 상세페이지로 이동

9. 오픈API상세 페이지에서 **관련기술문서 링크**를 웹페이지에 입력하여 해당 문서를 다운로드함

2. API 사용하기

실습예제1. '연도별 부동산 거래건수 조회' 데이터 크롤링

1) **[부동산 거래현황 통계조회서비스]** 기술문서를 참고하여 **api 요청/응답 테스트**

- '24 연도별 부동산 거래 건수 조회'를 찾아서 요청메세지 예제 코드를 복사하여 웹브라우저 주소창에 넣고 서비스 키를 뒤에 붙여 클릭하여 응답 메시지 결과대로 나오는지 확인

■ c) 응답 메시지 명세

항목명(영문)	항목명(국문)	항목크기	항목구분	샘플데이터	항목설명
<u>research_date</u>	조사일자	6	1	2021	조사일자
<u>region_cd</u>	지역코드	10	1	11000	지역코드
<u>deal_obj</u>	거래유형	2	1	01	거래유형
<u>all_cnt</u>	합계_건수	10	1	295860	합계_건수
<u>region_nm</u>	지역명	50	1	서울	지역명
<u>level_no</u>	지역구분 레벨	1	1	0	지역구분 레벨

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

↵

■ d) 요청 / 응답 메시지 예제

요청메시지
https://api.odcloud.kr/api/RealEstateTradingSvc/v1/getRealEstateTradingCountYear?page=1&perPage=1&cond%5BRESEARCH_CD%3A%3AEQ%5D=11000&cond%5BRESEARCH_DATE%3A%3ALTE%5D=2021&cond%5BRESEARCH_DATE%3A%3AGTE%5D=2021&cond%5BDEAL_OBJ%3A%3AEQ%5D=01&serviceKey=서비스키
응답 메시지
<pre>{ "currentCount": 1, "data": [{ "ALL_CNT": 295860, "DEAL_OBJ": "01", "LEVEL_NO": "0", "REGION_CD": "11000",</pre>

2. [부동산 거래현황 통계조회서비스] 페이지에서 아래로 스크롤하여 '활용신청 상세기능정보' 항목

- 모든 부동산 정보가 나열되어 있음
- '연도별 부동산 거래 건수 조회'를 클릭하여 입력 매개변수와 값을 확인

In [1]:

```
# 요청페이지 url+요청인자(파라미터)

endpoint='https://api.odcloud.kr/api/RealEstateTradingSvc/v1/getRealEstateTradingCountYear?'
key='tZvhTFt07H3kNWZxjEFGZsGmVILjG5XD33L920MvnShzaFrXQGUY89fLk0MMWn4%2BspA2l78%2FHiRNzaF6lw3mdw%3D'

page = '1'
perPage = '2'
region_cd = '11000' # 전국
start = '2020'
end = '2021'
deal_obj = '05' # 토지 아파트:05

params = 'page='+ page +'&'W
'perPage='+perPage+'&'W
'cond[REGION_CD::EQ]='+ region_cd +'&'W
'cond[RESEARCH_DATE::LTE]='+ end +'&'W
'cond[RESEARCH_DATE::GTE]='+ start +'&'W
'cond[DEAL_OBJ::EQ]='+ deal_obj +'&'W
'serviceKey='+key
```

In [2]:

```
url = endpoint + params
url
```

Out[2]:

```
'https://api.odcloud.kr/api/RealEstateTradingSvc/v1/getRealEstateTradingCountYear?
page=1&perPage=2&cond[REGION_CD::EQ]=11000&cond[RESEARCH_DATE::LTE]=2021&cond[RESE
ARCH_DATE::GTE]=2020&cond[DEAL_OBJ::EQ]=05&serviceKey=tZvhTFt07H3kNWZxjEFGZsGmVILj
G5XD33L920MvnShzaFrXQGUY89fLk0MMWn4%2BspA2l78%2FHiRNzaF6lw3mdw%3D%3D'
```

```
https://api.odcloud.kr/api/RealEstateTradingSvc/v1/getRealEstateTradingCountYear?
page=1&
perPage=1&
cond[REGION_CD::EQ]=11000&
cond[RESEARCH_DATE::LTE]=2021&
cond[RESEARCH_DATE::GTE]=2021&
cond[DEAL_OBJ::EQ]=01&
serviceKey=서비스키
```

[실습코드] BeautifulSoup을 이용한 크롤링

In [3]:

```
import requests
from urllib.request import urlopen
from bs4 import BeautifulSoup
import pandas as pd
```

크롤링할 API 요청하기

- 2019년~2021년 종로구 아파트거래현황 데이터

In [4]:

```
endpoint='https://api.odcloud.kr/api/RealEstateTradingSvc/v1/getRealEstateTradingCountYear'
key=''
```

In [5]:

```
page = '1'
perPage = '5'
region_cd = '11110' # 종로구
start = '2019'
end = '2021'
deal_obj = '05' # 아파트:05
```

In [6]:

```
params = '?page='+ page + '&'W
'perPage='+perPage+'&'W
'cond[REGION_CD::EQ]='+ region_cd + '&'W
'cond[RESEARCH_DATE::LTE]='+ end + '&'W
'cond[RESEARCH_DATE::GTE]='+ start + '&'W
'cond[DEAL_OBJ::EQ]='+ deal_obj + '&'W
'serviceKey='+key
```

In [7]:

```
url = endpoint + params
```

In [8]:

```
response = urlopen(url)
```

In [9]:

```
response.read()
# json 형식
```

Out[9]:

```
b'{"currentCount":3,"data":[{"ALL_CNT":838,"DEAL_OBJ":"05","LEVEL_NO":"1","REGION_CD":"11110","REGION_NM":"WxecWxa2Wx85WxebWxa1Wx9cWxeaWxb5Wxac","RESEARCH_DATE":"2021"},{"ALL_CNT":1181,"DEAL_OBJ":"05","LEVEL_NO":"1","REGION_CD":"11110","REGION_NM":"WxecWxa2Wx85WxebWxa1Wx9cWxeaWxb5Wxac","RESEARCH_DATE":"2020"},{"ALL_CNT":966,"DEAL_OBJ":"05","LEVEL_NO":"1","REGION_CD":"11110","REGION_NM":"WxecWxa2Wx85WxebWxa1Wx9cWxeaWxb5Wxac","RESEARCH_DATE":"2019"}],"matchCount":3,"page":1,"perPage":5,"totalCount":34244}Wn'
```

실습예제2. 부동산 거래건수 데이터 크롤링

2019년1월~2022년6월까지 종로구 아파트거래 현황 데이터 수집

XML 형식으로 가져오기

- 부동산거래건수 조회 API 사용 : 월별거래건수 조회가능

- API이름 : '/RealEstateTradingSvc/v1/getRealEstateTradingCount'

In []:

```
endpoint = 'https://api.odcloud.kr/api/RealEstateTradingSvc/v1/getRealEstateTradingCount'
```

In []:

```
returnType='XML'  
page = '1'  
perPage = '42'  
region_cd = '11110' # 종로구  
start = '201901'  
end = '202206'  
deal_obj = '05' # 아파트:05
```

In []:

```
url = endpoint + params + '&returnType=' + returnType
```

- urllib.request.urlopen() 사용

In []:

```
res = urlopen(url)  
soup = BeautifulSoup(res, 'xml')
```

In []:

```
soup
```

- requests.get()사용

In []:

```
params_dic = {'returnType':returnType,  
              'page': page,  
              'perPage': perPage,  
              'cond[REGION_CD::EQ]':region_cd,  
              'cond[RESEARCH_DATE::LTE]':end,  
              'cond[RESEARCH_DATE::GTE]':start,  
              'cond[DEAL_OBJ::EQ]':deal_obj,  
              'serviceKey':key,  
              }
```

In []:

```
res = requests.get(endpoint, params=params_dic)  
res
```


In []:

부동산거래 현황을 데이터프레임으로 저장

In []:

```
items = soup.find_all('item')
items
```

- item 태그 하나에 들어있는 정보를 딕셔너리로 저장하는 함수 작성

In []:

```
columns = ["REGION_NM", "RESEARCH_DATE", "ALL_CNT", "DEAL_OBJ", "LEVEL_NO", "REGION_CD"]
estate = {}
```

In []:

```
def to_real_estate(item_tag):
    columns = ["REGION_NM", "RESEARCH_DATE",
               "ALL_CNT", "DEAL_OBJ",
               "LEVEL_NO", "REGION_CD"]
    estate = {}
    for col in columns:
        estate[col] = item.find('col', {'name': col}).text
    return estate
```

In []:

```
estate_list = []
for item in items:
    estate = to_real_estate(item)
    estate_list.append(estate)
estate_list
```

- 조회한 부동산거래 전체 데이터를 읽고 데이터프레임으로 저장

In []:

```
df = pd.DataFrame(estate_list,
                  columns=columns)
```

In []:

```
df
```

과제. 공중화장실개방현황 데이터를 API로 수집

- 경기데이터드림 사이트 : <https://data.gg.go.kr/portal/mainPage.do>
(<https://data.gg.go.kr/portal/mainPage.do>)
 - 데이터 다운로드 링크 주소 <https://data.gg.go.kr/portal/data/service/selectServicePage.do?infId=GW6U772M6045H11Q799612585601&infSeq=1>
(<https://data.gg.go.kr/portal/data/service/selectServicePage.do?infId=GW6U772M6045H11Q799612585601&infSeq=1>)
1. 수집한 데이터를 데이터프레임으로 저장
 2. 데이터 전처리를 진행하여 화장실의 위치를 지도로 시각화
 - 중복데이터, 결측치, 주소 변경 등(실습평가문제 참고)
 3. 작성한 파일을 구글드라이브에 업로드
 - 구글드라이브 링크:
<https://drive.google.com/drive/folders/15CQWv99PU2LMJQCn4UqcfDvPJK49UDQS?usp=sharing>
(<https://drive.google.com/drive/folders/15CQWv99PU2LMJQCn4UqcfDvPJK49UDQS?usp=sharing>)
-