

2022 주한 베트남 유학생 SW교육 프로그램



문서명	지정과제
교육생	NGUYEN DINH HUONG (응웬딩흐엉)
작성일	2022.08.16
강사님	Java(김창호), Cloud/Elastic Search(정태성) Log Stash(박태석), Kibana (최승용)
Github	https://github.com/huonghope/sw-elk-2022 (본과제에서 작성된 코드 저장소)
Kibana URL	http://20.196.220.19:5601/

< 목차 >

1. 공공데이터 회원가입	4
2. 환경구축	5
2.1 배경지식 및 Infrastructure 환경 준비	5
가. 배경지식	5
나. Infrastructure 환경 준비	8
2.2 Elastics 설치	11
2.3 Logstash 설치	12
2.4 Kibana 설치	13
3. 과제선정	14
3.1. 전기차충전 데이터	15
3.2. 유기견보호 데이터	29
3.3. 한우 낙농 . 축산물품질평가원 축산물경락가격정보 데이터	30
3.4. 농경락	31
4. 가점사항	14
4.1. 공공데이터의 API 를 이용하여 동적으로 Logstash 연동	34
4.2. 지정과제에서 명시된 외의 데이터를 참조하여 시각화	34
4.3. MySQL 같은 RDB 연동로 Logstash 연동	35
4.4. Crawling 같은 제의 3방법을 통해서 데이터 취득	36

1, 공공데이터 회원가입

공공데이터포털은 행정안전부에서 운영하는 공공데이터 통합제공 시스템이다. 대한민국 정부가 보유한 다양한 공공데이터를 개방하여 누구나 편리하고 손쉽게 활용할 수 있게 하는 것을 목적으로 한다. 행정안전부 공공데이터정책과에서 관련 정책을 추진하고 있다.

공공데이터포털 웹 사이트 <https://www.data.go.kr/> 주소를 접속해서 회원가입을 진행하였다. 공공데이터 포털에서 얻을 수 있는 데이터는 크게 3종류로 나누어져 있습니다. 첫째, 파일 데이터이며 이는 직접 다운로드 받아 사용할 수 있는 데이터이다. 예를 들어 워드 문서나 엑셀 같은 파일 형태의 데이터이다. 둘째, 오픈 API 데이터이며 오픈 API는 XML 로 만들어져 있다. 사용하는 앱들은 OPEN API를 연동하여 이를 사용하고 IT 기술자들이 주로 사용하는 것으로 사람이 직접보고 사용하기에는 다소 무리가 있는 형태이다. 오픈API를 사용하기 위해서는 개발자 키를 받아야 하며 이 키를 사용하여 인증을 하고 데이터를 주고 받을 수 있다. 해당 방법은 다음과 같이 더 자세히 설명할 것이다. 셋째 시각화 데이터이며 바로 눈으로 확인 할 수 있으며, 조회 조건을 주어 세부적으로 검색을 할 수도 있다. 웹 에서 테이블 형식으로 나오기 때문에 데이터를 즉시 파악할 수 있고 또한 행의 상단을 클릭하면 정렬이 가능하며 우측에 조회조건을 주어 검색이 가능한 등 각종 편의 기능들을 제공하고 있다. [표 1]는 공공데이터포털에서 제공한 데이터 종류를 보여주고 있다.

표 1. 공공데이터포털 제공 데이터 종류

공공데이터포털 제공 데이터 종류			
데이터 타입	파일	Open API	시각화
데이터 형식	CSV, JSON + XML, HWP, WORD	JSON, XML	웹 UI
특징	파일로 쉽게 접근한 장점	API 키를 필요함	바로 확인할 수 있음

본 과제에서 파일 타입 및 OPEN API 타입을 사용해서 데이터를 받아고 분석할 것이다. 파일 타입인 경우에는 사용자는 원하는 데이터를 검색해서 조회된 결과에서 파일데이터를 선택해서 파일을 다운로드하면 된다. 데이터 상세 페이지에서 파일데이터 정보를 여러 가지 확인할 수 있으며 파일명, 분류체계 관리부서명, 확정자 등과 같은 정보가 있다. 또는 데이터를 바로 확인 할 수 있도록 미리보기도 제공한다. 파일데이터의 장점은 기술 개발자가 아니더라도 데이터를 쉽게 접근할 수 있고 파악할 수 있는 점이다. 다만 받고 자하는 데이터 파일의 용량이 큰 경우에는 저장소를 주의해야 한다. [그림1]는 파일 데이터 정보를 출력하는 화면 예제를 보여주고 있다.

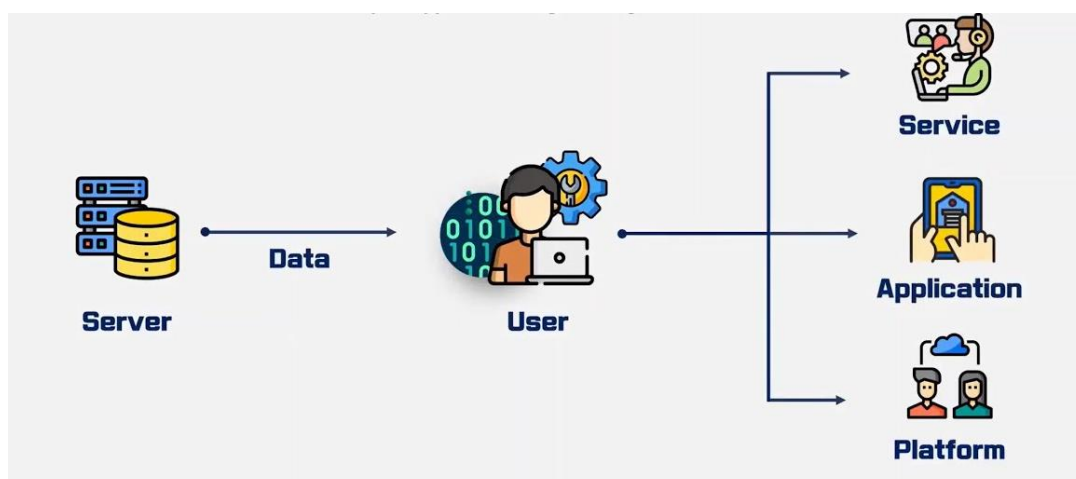
그림1 파일데이터 정보

파일데이터 정보			
메타데이터 다운로드			
파일데이터명	충청북도 보은군_코로나 월별 현황_20220207		
분류체계	사회복지 - 보건의료	제공기관	충청북도 보은군
관리부서명	행정과	관리부서 전화번호	043-540-3124
보유근거		수집방법	해당부서에 데이터 요청
업데이트 주기	수시 (1회성 데이터)	차기 등록 예정일	
매체유형	텍스트	전체 행	14
확장자	CSV	다운로드(바로그기)	50
데이터 한계		키워드	코로나, 확진자 수, 사망자 수
등록	2022-02-07	수정	2022-02-07
제공형태	공공데이터포털에서 다운로드(원문파일등록)		
설명	충청북도 보은군 코로나(보은군, 월별, 확진자 데이터, 사망자 데이터) 월별 현황입니다. 확진자 수 와 사망자 수에 대한 월별 데이터입니다.		
기타 유의사항			
비용부과유무	무료	비용부과기준 및 단위	건
이용허락범위	이용허락범위 제한 없음		

출처: 공공데이터포털

API는 Application Programming Interface(애플리케이션 프로그램 인터페이스)의 줄임말이며 정의 및 프로토콜 집합을 사용하여 두 소프트웨어 구성 요소가 서로 통신할 수 있게 하는 메커니즘이다. 공공데이터API란 나라에서 보유한 데이터 프로그램을 허가를 받아 각자의 인증키로 사용할 수 있는 것이다. OPEN API 타입인 경우에는 기술 개발자들을 많이 활용하고 있으며 데이터 다운로드 자동화 할때 사용하고 있다. 다만, 기술 개발자가 아닌 사용자라면 처음에 데이터를 접근할 때 어려움을 겪을 수 다. [그림2]는 공공데이터 포털 OPEN API 사용 흐름을 보여주고 있다.

그림2. Open Application Programming Interface



OPEN API의 인증키를 받으려면 먼저 회원가입을 진행해야 한다. 회원가입을 한 후에 받고 자하는 데이터를 검색해서 조회된 결과에서 오픈 API를 선택한 후에 활용신청을 클릭하면 된다. 활용신청 단계에서 몇가지 간단한 정보를 입력하면 데이터 활용 승인되며 인증키를 바로 확인할 수 있다. [그림3]는 OPEN API의 인증키 예제를 보여주고 있다.

그림3. OPEN API의 인증키

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)	1oRqU9tV%2FmfK1hpdUgBtBlxWSzmlAd4juPaiCvrxXckTDzh6k5mir9NI9fEBgFJ0D1AYNySFppcB0ZJfiG2IA%3D%3D
일반 인증키 (Decoding)	1oRqU9tV/mfK1hpdUgBtBlxWSzmlAd4juPaiCvrxXckTDzh6k5mir9NI9fEBgFJ0D1AYNySFppcB0ZJfiG2IA==

OPEN API를 사용하는 경우에는 공공데이터포털에서 API를 호출하기 위해서 여러 프로그래밍 언어 예제 소스코드도 제공한다. 이를 통해서 기술 개발자를 처음 개발 접근하는 데에 도움이 될 것이다. 해당 기능을 본 과제에서도 진행해보도록 한다.

2, 환경구축 (기초 Infra 설치)

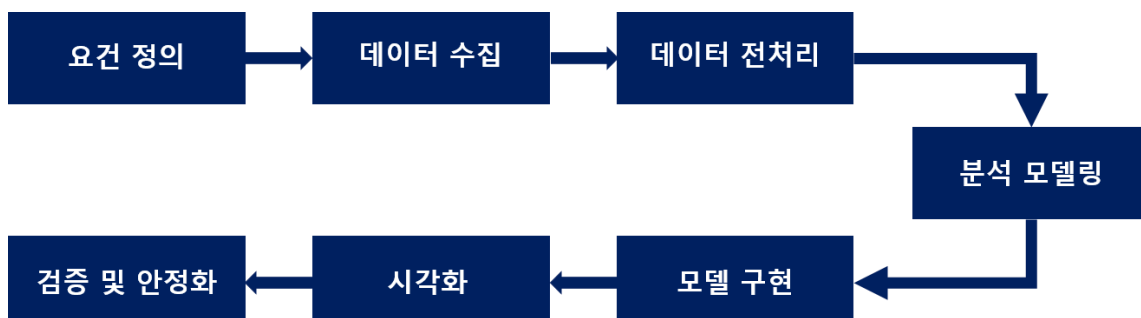
2.1 배경지식 및 Infrastructure 환경 준비

가. 배경지식

빅데이터는 기존의 방법으로 처리가 어려운 대용량 혹은 다양한 유형의 실시간 생성되는 데이터의 집합을 의미한다. 빅데이터 분석은 시간, 공간 그 외 조건들의 변화에 따른 데이터의 분포와 데이터 간의 상호관계를 다양한 관점으로 조망하므로 의미 있는 패턴을 발견하고 세상을 이해해 가는 과정이다. 빅데이터는 일상생활 여러 분야에서 활용하고 있으며 맞춤형 서비스 분야, 국민 안전 분야, 조세 분야, 의료·복지 분야, 교통·환경 분야, 교육 분야 등과 같은 대표적인 분야다.

빅데이터를 추진 절차는 보통 7단계를 거치며 요건 정의, 데이터 수집, 데이터 전처리, 분석 모델링, 모델 구현, 시각화, 검증 및 안정화이다. 그림4는 빅데이터를 추진 절차 7단계를 보여주고 있다.

그림4. 빅데이터를 추진 절차



- **요건 정의 단계**는 빅데이터 분석과 활용을 통해 해결하고자 하는 현안에 대해 명확히 이해하고 정의하며, 사업 추진을 위해 필요한 환경을 분석 하고 사업의 추진 계획을 수립하는 과정을 포함한다. 이는 사업의 기본적인 이해를 바탕으로 현업 및 IT 담당자와 인터뷰를 수행하고 현안을 분석하며 관련 자료를 수집, 정리하여 요구사항을 정의한다.
- **데이터 수집 단계**에서는 분석 데이터를 정의하고 수집 환경을 구성하여 데이터를 수집한다. 분석 데이터 정의 단계에서는 분석 범위에 맞는 데이터 수집 대상 목록을 정리하고 보유기관 및 담당자를 조사한다. 수집 환경 구성 단계에서는 기관이 보유한 수집 환경이나, 빅데이터 포털 인프라를 활용하여 데이터를 수집하고 이를 위한 환경을 구성한다. 데이터 수집 단계에서는 수집 대상을 명확하게 정의하여 보유기관에 요청하여 수집한다.
- **데이터 전처리 단계**는 수집된 데이터를 파악하여 데이터의 오류를 점검하고, 데이터를 정제함과 동시에 매칭 key 값을 기준으로 여러 데이터를 융합하고 기초 통계 분석을 통해 데이터의 고유 패턴을 파악한다.
- **분석모델링 단계**에서는 분석 대상 및 범위를 정하여 분석 목적을 구현하기 위한 분석 방법론을 설계하는 단계다. 이는 타당한지 자문위원회 회의를 거쳐 논의하고, 내용을 추가 반영하여 분석모델 설계를 완성한다
- **모델 구현 단계**에서는 분석모델링 설계를 알고리즘으로 구현하기 위해 알고리즘 모델링을 설계하고 모듈을 개발하여 분석 알고리즘을 구현한다.
- **시각화 단계**에서는 분석 결과가 단순 수치가 아닌, 분석가 이외의 사람들이 이해하기 쉽도록 그림 및 도표를 활용하여 표현하며 수집한 데이터 및 분석 결과를 담당자 및 관련 기관들과 공유하기 위한 목적으로 필요하다면 Open-API 형태로 제공하도록 개발한다.
- **검증 및 안정화 단계**에서는 사업 전반에 대한 결과물(시각화, 분석 프로세스 자동화 등)을 통합 테스트하여 검증하고, 사용자 테스트 후 검수를 거쳐 분석 결과를 사용할 실무자를 위하여 교육을 실시하고 인수인계를 한다.

빅데이터의 5가지 구성요소를 포함하며 규모, 다양성, 속도, 정확성 및 가치가 있다. 빅데이터 분석기법이 여러 가지 있는데 대표적인 기법은 기계학습, 데이터마이닝, 테스트마이닝, 소셜네트워크분석 등이 있다.

표 2. 빅데이터 분석기법

빅데이터 분석기법	내용
기계학습 기법	방대한 데이터를 분석해 미래를 예측하는 기술로 일반적으로 생성(발생)된 데이터를 정보와 지식(규칙)으로 변환 하는 컴퓨터 알고리즘(algorithm)을 의미한다.

데이터마이닝 기법	축적된 대용량 데이터를 통계기법 및 인공지능기법을 이용하여 분석하고 이에 대한 평가를 거쳐 일반화시킴으로써 새로운 자료에 대한 예측 및 추측을 할 수 있는 의사결정을 지원한다. 데이터마이닝은 분류(classification), 추정(estimation), 예측(prediction), 유사집단화(affinity grouping), 군집화(clustering)의 5가지 업무영역을 수행한다.
텍스트마이닝	텍스트마이닝은 자연어(natural language) 형태로 구성된 비정형 또는 반정형 데이터에서 패턴 또는 관계를 추출하여 의미 있는 정보를 찾아내는 기법으로 자연어처리(natural language processing, NLP)가 핵심기술이다.
소셜네트워크분석	소셜네트워크분석(Social Network Analysis: SNA)은 소셜시스템(social system)이 관계(relationship)와 이러한 관계에 의해 형성되는 패턴(pattern)에 의해 창조된다는 전제로 한다.

빅데이터는 통상적으로 사용되는 데이터 수집, 관리 및 처리 소프트웨어의 수용 한계를 넘어간다. 이로 인해서 분석, 처리 등과 같은 작업을 시간이 많이 걸린다. 이 문제를 해결해서 자동 수집, 시각화 등과 같은 도구나 오픈소스를 존재한다.

이번 수업에서 언급한 ELK는 3가지 오픈소스 소프트웨어 Elastic Search, LogStatsh, Kibana의 조합을 말하며 각 제품이 연동되어 데이터 수집 및 분석 툴로서 동작한다. 추가로 ELK Stack는 ELK 솔루션에서 Beats 추가하는 작업을 의미하며 Beats는 서버에 에이전트로 설치하여 다양한 유형의 데이터를 ElasticSearch 또는 Logstash에 전송하는 오픈소스 데이터 발송자다.

그림5. ELK별로 주로 수행 작업



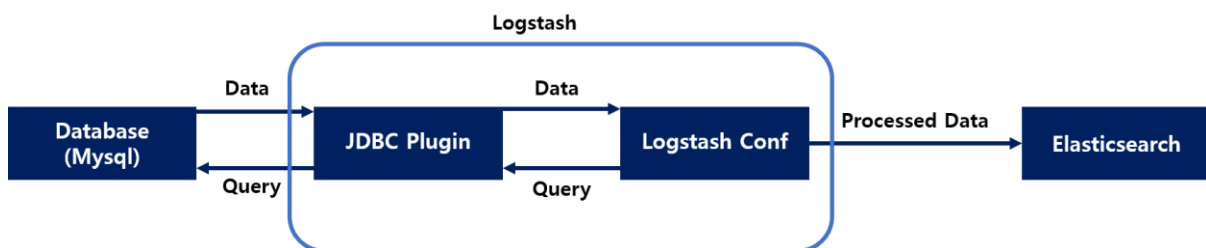
출처: medium

- **ElasticSearch**는 Lucene 기반으로 개발한 분산 검색엔진으로, Logstash를 통해 수신된 데이터를 저장소에 저장하는 역할을 담당한다. 데이터를 중심부에 저장하여 예상되는 항목을 검색하고 예상치 못한 항목을 밝혀낼 수 있다. 정형, 비정형, 위치정보, 메트릭 등 원하는 방법으로 다양한 유형의 검색을 수행하고 결합할 수 있다.

- **Logstash**는 오픈소스 서버측 데이터 처리 파이프라인으로, 다양한 소스에서 동시에 데이터를 수집하고 변환하여 **stash** 보관소로 보낸다. 수집할 로그를 선정해서, 지정된 대상 서버(ElasticSearch)에 인덱싱하여 전송하는 역할을 담당하는 소프트웨어이다.
- **Kibana**는 Elasticsearch 인덱스에 저장된 로그 데이터를 검색, 분석 및 시각화하기 위한 웹 기반 그래픽 인터페이스이다. Elasticsearch의 REST 인터페이스를 활용하여 데이터를 검색하고 사용자가 데이터에 대한 맞춤형 대시보드 보기를 생성할 수 있을 뿐만 아니라 임시 방식으로 데이터를 쿼리 및 필터링할 수 있다.

엘라스틱 스택(구, ELK스택)은 비정형 혹은 정형 데이터를 가리지 않고 통합해서 검색하고 분석하기 위한 풀스택을 제공하기 때문에 최근에는 기존의 데이터베이스에 저장된 데이터 중에서 검색 및 분석을 하고 싶은 데이터셋만 엘라스틱 서치로 인덱싱하고 키바나로 자유롭게 분석하고 리포팅을 만드는 유스케이스가 많아졌다. 기존에 각종 트랜잭션 데이터, 로그, 메트릭, 보안 이벤트 데이터 등도 오라클이나 MySQL에서 관리하는 것보다 이러한 비정형 데이터들은 엘라스틱 서치로 인덱싱함으로써 검색 및 분석을 더 용이하게 하는 형태로 많이 활용이 되고 있다. [그림6]은 데이터베이스 및 Elasticsearch를 Logstash를 통해서 데이터 전송할 때 흐름을 보여주고 있다.

그림6. 데이터베이스 및 Elasticsearch 통신 흐름



출처: blog

나. Infrastructure 환경 준비

본 과제에서 Microsoft Azure에서 제공하는 Ubuntu VM를 사용해서 개발 환경을 구축하고 자한다. 각 도구들을 설치 완료한 후에 설정 파일을 통해서 도구를 연결할 수 있고 외에 접근할 수 있다. 등록된Ubuntu VM가 같아 환경정보는 다음과 같다.

표 3. Microsoft Azure에서 제공하는 VM 정보

운영체제	Ubuntu Server
버전	20.04 LTS – Gen2
메모리	8 GiB
제공 기관	Microsoft Azure
vCPUs	2

Microsoft Azure에서 제공하는 Ubuntu VM를 등록절차를 다음과 같다.

1. Microsoft Azure 회원가입
2. 로그인한 다음에 Azure Service답에서 Virtual Machines를 선택하고 Create -> Azure virtual machine를 클릭하면 VM정보를 입력 페이지를 이동된다.
3. VM 정보를 입력 페이지에서 필요하는 정보를 입력한 다음에 Create를 클릭하면 VM를 생성 완료된다. [그림8]VM 정보를 입력페이지를 보여주고 있다.
4. 본 과제에서 생성되는 VM 기본 정보를 위에 테이블을 확인 할 수 있다. 또는 웹 사이트에서 여러 가지 정보를 확인도 할 수 있다.

그림7. VM 정보

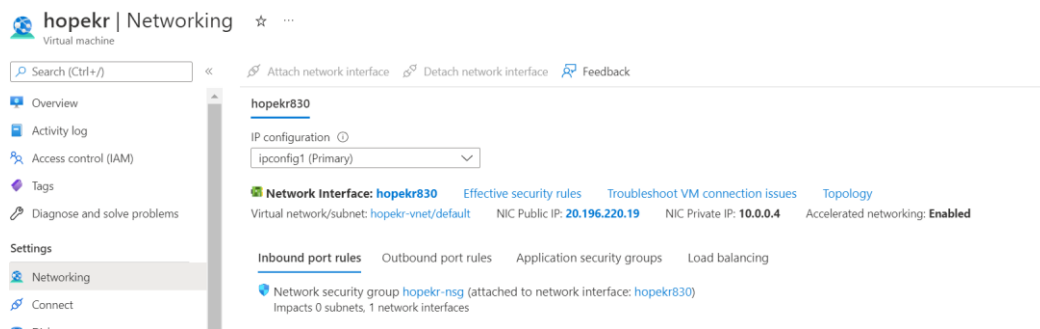


그림8. VM 정보를 입력 페이지

Create a virtual machine ...

Basics Disks Networking Management Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)


i This subscription may not be eligible to deploy VMs of certain sizes in certain regions.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Azure for Students
 Resource group * ⓘ (New) hopekr2_group
[Create new](#)

Instance details

Virtual machine name * ⓘ hopekr2 ✓
 Region * ⓘ (Asia Pacific) Korea Central ✓
 Availability options ⓘ No infrastructure redundancy required ✓
 Security type ⓘ Standard ✓
 Image * ⓘ  Ubuntu Server 20.04 LTS - Gen2 ✓
[See all images](#) | [Configure VM generation](#)
 Run with Azure Spot discount ⓘ ☐
 Size * ⓘ Standard_D2s_v3 - 2 vcpus, 8 GiB memory (\$89.79/month) ✓
[See all sizes](#)

Administrator account

Authentication type ⓘ ☐ SSH public key
☒ Password
 Username * ⓘ hopekr ✓
 Password * ⓘ ✓
 Confirm password * ⓘ ✓

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ ☐ None
☒ Allow selected ports
 Select inbound ports * ⓘ SSH (22) ✓

⚠ This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

[Review + create](#)

[< Previous](#)

[Next : Disks >](#)

2.2 Elasticsearch 설치

Elasticsearch를 설치하기 위해서 먼저 Java를 설치해야 한다. (Java 8 이상)

```
# Ubuntu package 업데이트
sudo apt-get update

# Java JDK 설치
sudo apt-get install openjdk-11-j

# 설치 성공되었으면 버전을 확인
java -version

# 환경변수 추가 (참고로 profile나 ~/.bashrc에서 추가 권장)
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export PATH=$JAVA_HOME/bin:$PATH
export CLASS_PATH=$JAVA_HOME/lib:$CLASS_PATH

# Elasticsearch 서비스를 동작 확인하기 위해서 포트를 세팅
sudo ufw disable
sudo iptables -F
sudo apt-get install iptables-persistent netfilter-persistent
sudo apt install net-tools

# Elasticsearch 기본 포트:9200 / Kibana 기본 포트: 5601
sudo iptables -A INPUT -p tcp --dport 9200 -j ACCEPT

# 포트 설정 확인
netstat -nap

# Ubuntu package 다시 업데이트
sudo apt-get update

# 저장소에 접근 허락해주는 package 설치
sudo apt install apt-transport-https

# 패키지 리포지토리에서 설치
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
sudo sh -c 'echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" >
/etc/apt/sources.list.d/elastic-7.x.list'

# Ubuntu package 다시 업데이트
sudo apt update
```

```
# Elasticsearch 패키지 설치
sudo apt install elasticsearch

# Elasticsearch 서비스를 시작
sudo systemctl enable elasticsearch.service
sudo service elasticsearch start

# Elasticsearch 서비스를 종료
sudo service elasticsearch stop
```

외부에 접근하기 위해서 Elasticserach 설정 파일을 수정

```
sudo vi /etc/elasticsearch/elasticsearch.yml
```

설치 완료 확인 화면

```
/home/hopekr/upload > curl -XGET 'http://localhost:9200'
{
  "name" : "hopekr",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "SWRBrGV2RbqN69aL-geWyg",
  "version" : {
    "number" : "7.17.5",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "8d61b4f7ddf931f219e3745f295ed2bbc50c8e84",
    "build_date" : "2022-06-23T21:57:28.736740635Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

2.3 Logstash설치

Logstash를 설치하기 위해서 먼저 Java를 설치해야 한다. (Java 8 이상)

```
# Ubuntu package 업데이트
sudo apt-get update

# Java JDK 설치
sudo apt-get install openjdk-11-j

# 설치 성공되었으면 버전을 확인
java -version
```

```
# 환경변수 추가 (참고로 profile나 ./bashrc에서 추가 권장)
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export PATH=$JAVA_HOME/bin:$PATH
export CLASS_PATH=$JAVA_HOME/lib:$CLASS_PATH

# Ubuntu package 다시 업데이트
sudo apt-get update

# 저장소에 접근 허락해주는 package 설치
sudo apt install apt-transport-https

# 패키지 리포지토리에서 설치
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
sudo sh -c 'echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" >
/etc/apt/sources.list.d/elastic-7.x.list'

# Ubuntu package 다시 업데이트
sudo apt update

# Logstash 패키지 설치
sudo apt install logstash

# Logstash 서비스를 시작
sudo systemctl enable logstash
sudo service logstash start

# Logstash 서비스를 종료
sudo service logstash stop
```

설치 완료 확인 화면

```
/home/hopecr/upload > /usr/share/logstash/bin/./logstash --version
Using bundled JDK: /usr/share/logstash/jdk
logstash 7.17.5
/home/hopecr/upload >
```

2.4 Kibana설치

Kibana를 설치하기 위해서 먼저 Java를 설치해야 한다. (Java 8 이상)

```
# Ubuntu package 업데이트
sudo apt-get update

# Java JDK 설치
sudo apt-get install openjdk-11-j

# 설치 성공되었으면 버전을 확인
java -version
```

```
# 환경변수 추가 (참고로 profile나 ./bashrc에서 추가 권장)
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export PATH=$JAVA_HOME/bin:$PATH
export CLASS_PATH=$JAVA_HOME/lib:$CLASS_PATH

# Kibana서비스를 동작 확인하기 위해서 포트를 세팅
sudo ufw disable
sudo iptables -F
sudo apt-get install iptables-persistent netfilter-persistent
sudo apt install net-tools

# Kibana 기본 포트: 5601
sudo iptables -A INPUT -p tcp --dport 5601 -j ACCEPT

# 포트 설정 확인
netstat -nap

# Ubuntu package 다시 업데이트
sudo apt-get update

# 저장소에 접근 허락해주는 package 설치
sudo apt install apt-transport-https

# 패키지 리포지토리에서 설치
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
sudo sh -c 'echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" >
/etc/apt/sources.list.d/elastic-7.x.list'

# Ubuntu package 다시 업데이트
sudo apt update

# Kibana 패키지 설치
sudo apt install logstash

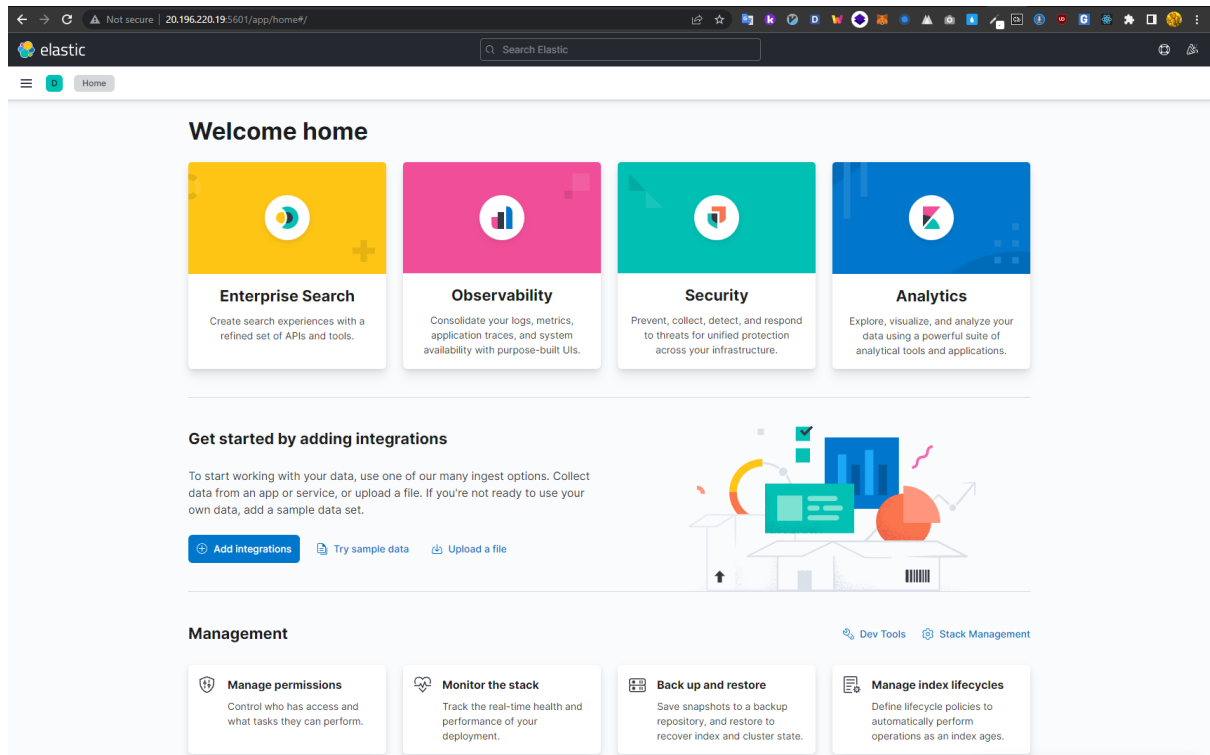
# Kibana 서비스를 시작
sudo systemctl enable kibana
sudo service kibana start

# Logstash 서비스를 종료
sudo service kibana stop
```

외부에 접근하기 위해서 Kibana 설정 파일을 수정

```
sudo vi /etc/kibana/kibana.yml
```

설치 완료한 다음에 Kibana 접속 화면



3, 과제 선정

공공데이터 포털 오픈 API를 활용해서 데이터셋을 수집 방법

1. 공공데이터 포털 접속
2. 회원가입 및 로그인
3. 원하는 데이터 검색
4. 해당 데이터를 맞는지 확인 (제목 및 제공기관) 및 “활용신청” 클릭
5. 활용신청이 완료되면 인코딩 및 디코딩 인증기가 제공되며 마이페이지에서 오픈 API - > 개발과정에서 요청 데이터 상세보기 페이지에서 확인이 가능함
6. 활용신청 완료된 데이터 상세페이지에서 첨부된 참고 문서 (docs, hwp)를 다운로드하고 읽어보며 인증키가 등록될 때까지 기다림 (인증키 자체를 발급되지만 공공데이터 포털쪽에 등록이 되어야만 사용이 가능하며 보통 1시간이 걸리거나 더 걸릴 수 있다.
7. 인증키를 정상적으로 등록이 완료되면 요청 데이터 상세페이지에서 키를 사용해서 API를 호출할 수 있는데, 기술 개발자 사용하는 언어에 따라 코드를 다를 수 있겠지만 예제코드도 제공해서 참고하면 도움이 될 것이다. (인증키를 정상적으로 등록되지 않으면 ServiceKey is Not Registered Error를 출력됨)

3.1. 전기차충전: 한국전력공사의 오픈 API 를 활용하여 충전소 위치 검색 실시간 충전

기 모니터링 다크모드 지원 충전기 종류 확인 충전소 검색 기능 충전소 오류 제보 등의 정보를 제공합니다.

- 가) 한국환경공단_전기자동차 충전소 정보([링크](#))
- 나) 한국전력공사_전기차 충전소 운영정보([링크](#))

개요

전기차는 고전압 배터리에서 전기에너지를 전기모터로 공급하여 구동력을 발생시키는 차량으로, 화석연료를 전혀 사용하지 않는 무공해 차량이다. 의 전기차 누적 보급대수는 2021년 9월 말 기준 약 20만대로 전체 등록된 자동차 대수의 0.8%이다. 하지만 전기차 충전기의누적 보급대수는 2021년 6월 기준 약 7만 2000기이고, 그중 급속충전기는 1만 3000기, 완속 충전기는 5만 9000기이다. 이로 인해서 충전소의 부족하는 문제점을 발생한다. 전기차 충전소는 전기를 연료로 쓰는 전기자동차 전용 충전소이며 기존 유형 위주의 액체 연료 보급 시설인 주유소와 기체 연료 보급 시설인 수소충전소, LPG 충전소, CNG 충전소와 달리 무형의 전기를 끌어와 사용하는 형태이기에 타 보급 시설과 비교하여 전자기기 충전과 같은 느낌에 더 친숙하다. 가장 흔히 보급 된 전기자동차 형태인 "배터리 전기 자동차(BEV, Battery Electric Vehicle)"가 주로 사용하나, "플러그인 하이브리드 자동차(PHEV, Plug-in Hybrid Electric Vehicle)"도 사용하며, 수소연료전지차는 전기자동차이긴 하나 수소를 연료로 사용하기에 전기차 충전소가 아닌 수소충전소에서 충전한다. 충전소의 대표적인 종류는 고정형 충전소, 이동식 충전소, 충전요금 등이 있다. 전기차를 사용 문제중에서 충전소를 큰 하나 문제로써 충전서 운영정보 등관 같은 데이터를 분석할 만하다 생각한다.

데이터셋

가) 한국환경공단_전기자동차 충전소 정보

제공기관가 한국전력공사와 다른 한국환경공단이 발급한 데이터셋이다. 이 데이터셋 기준 정보를 다음과 같다.

데이터 형태	오픈 API	제공기관	한국환경공단
분류체계	환경 - 대기	데이터포맷	XML
API 유형	REST	키워드	전기차충전소 등
등록일자	2021-01-28	수정	2022-08-05
비용부과유무	무료	신청가능 트랙픽	
심의유형	개발단계: 자동승인 /운영단계: 자동승인		
이용허락범위	공공저작물_출처표시		

이전 2019년에 한국전력공사 기관은 제공한 데이터셋과 달라 정보가 많이 들어가 있다. 해당 데이터셋의 Docs 파일에서 응답 메시지 부분을 통해서 여러 정보를 받을 수 있다

는 것을 확인할 수 있다. 또는 각 데이터들이 표시된 코드나 설명도 같이 작성되어 있어 같이 확인할 수 있다.

그림10. API 호출 예시

③ 요청 메시지 명세

항목명(영문)	항목명(국문)	항목크기	항목구분	샘플데이터	항목설명
serviceKey	서비스키	100	1	인증키(URL Encode)	인증된 서비스키
pageNo	페이지 번호	5	1	1	조회 페이지 정수
numOfRows	한 페이지 결과 수	4	1	10	한 페이지 결과 수 (최소 10, 최대 9999)
zcode	지역구분 코드	2	0	11	시도 코드 (행정구역코드 앞 2자리)
zscore	지역구분 상세 코드	5	0	11680	지역구분 상세(행정구역코드 앞 2자리+법정동코드3자리)
kind	충전소 구분 코드	2	0	F0	충전소 구분
kindDetail	충전소 구분 상세 코드	4	0	F002	충전소 구분 상세
dataType	데이터 타입	4	1	JSON	데이터 타입(JSON, XML)

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

② 응답 메시지 명세

항목명(영문)	항목명(국문)	항목 크기	항목 구분	샘플데이터	항목설명
resultCode	결과코드	2	1	00	결과코드
resultMsg	결과메세지	50	1	NORMAL SERVICE.	결과메세지
totalCount	전체 건수	9	1	12000	요청 조건에 따른 전체 충전기 개수
pageNo	페이지 번호	5	1	1	조회 페이지
numOfRows	한 페이지 결과 수	5	1	10	한 페이지당 표출 데이터 수
items	목록		0..n		목록
statNm	충전소명	100	1	기후대기관	충전소명
statId	충전소ID	8	1	28260005	충전소ID
chgerId	충전기ID	2	1	02	충전기ID
chgerType	충전기타입	2	1	03	충전기타입 (01:DC차데모, 02: AC완속, 03: DC차데모+AC3상,

해당 데이터는 나. 오퍼레이션 목록은 다음과 같다.

일련 번호	서비스명(국문)	오퍼레이션명(영문)	오퍼레이션명(국문)	메시지명(영문)
1	전기자동차 충전소	getChargerInfo	충전기 정보 조회	INFO
2	정보 조회 서비스	getChargerStatus	충전기 상태 조회	STATUS

(1) 충전기 정보 조회 오퍼레이션 명세

해당 데이터셋을 OPEN API 실험 결과는 다음과 같다 (JAVA 언어)

요청 메시지 명세

항목명(영문)	항목명(국문)	값	항목설명
serviceKey	서비스키	100	인증된 서비스키
pageNo	페이지 번호	1	페이지 번호
numOfRows	한 페이지 결과 수	9999	한 페이지 결과 수 (최소 10, 최대 9999)
zcode	지역구분 코드	11 (서울특별시)	시도 코드 (행정구역코드 앞 2자리)

작성 코드

```
@Override
public void callOpenAPI() {
    // TODO Auto-generated method stub

    /*
     * 충전기 상태 조회
     * 전기자동차 보급 활성화 및 전기자동차 사용자를 위한 전국 공공 및 민간 충전기
     상태 실시간 제공(충전소ID, 충전기ID, 상태정보)
     */
    String urlGetChargerStatus =
"http://apis.data.go.kr/B552584/EvCharger/getChargerStatus";
    Boolean isGetChargerStatus = true;

    /*
     * 충전기 정보 조회
     * 전기자동차 보급 활성화 및 전기자동차 사용자를 위한 전국 공공 및 민간 충전기
     운영현황 실시간 제공(충전소명, 위치, 상태정보, 운영시간, 운영기관명, 충전기 용량)
     */
    String urlGetChargerInfo =
"https://apis.data.go.kr/B552584/EvCharger/getChargerInfo";

    StringBuilder urlBuilder = isGetChargerStatus ? new
StringBuilder(urlGetChargerStatus) :
        new StringBuilder(urlGetChargerInfo); /*URL*/
```

```

        try {
            urlBuilder.append("? " + URLEncoder.encode("serviceKey", "UTF-8")
+ "=" + Utils.getAPIKey()); /*Service Key*/
            urlBuilder.append("& " + URLEncoder.encode("pageNo", "UTF-8") + "=" +
URLEncoder.encode("1", "UTF-8")); /*페이지 번호*/
            urlBuilder.append("& " + URLEncoder.encode("numOfRows", "UTF-8") + "=" +
URLEncoder.encode("9999", "UTF-8")); /*한 페이지 결과 수 (최소 10, 최대 9999)*/
            urlBuilder.append("& " + URLEncoder.encode("period", "UTF-8") + "=" +
URLEncoder.encode("5", "UTF-8")); /*상태갱신 조회 범위(분) (기본값 5, 최소 1, 최대 10)*/
            urlBuilder.append("& " + URLEncoder.encode("zcode", "UTF-8") + "=" +
URLEncoder.encode("11", "UTF-8")); /*시도 코드 (행정구역코드 앞 2자리)*/
            URL url = new URL(urlBuilder.toString());
            HttpURLConnection conn = (HttpURLConnection) url.openConnection();
            conn.setRequestMethod("GET");
            conn.setRequestProperty("Content-type", "application/json");
            System.out.println("Response code: " + conn.getResponseCode());
            BufferedReader rd;
            if(conn.getResponseCode() >= 200 && conn.getResponseCode() <= 300) {
                rd = new BufferedReader(new
InputStreamReader(conn.getInputStream()));
            } else {
                rd = new BufferedReader(new
InputStreamReader(conn.getErrorStream()));
            }
            StringBuilder sb = new StringBuilder();
            String line;
            while ((line = rd.readLine()) != null) {
                sb.append(line);
            }
            rd.close();
            conn.disconnect();

            Document doc = Utils.createDocumentByURL(urlBuilder.toString());
            NodeList nList = doc.getElementsByTagName("item");
            System.out.println(nList.getLength());
            for(int temp = 0; temp < nList.getLength(); temp++){
                System.out.println("==== Item : " + (temp + 1) + " =====");
                Node nNode = nList.item(temp);
                Element eElement = (Element) nNode;
                map.put("busiId", Utils.getTagValue("busiId", eElement));
                map.put("statId", Utils.getTagValue("statId", eElement));
                map.put("chgerId", Utils.getTagValue("chgerId",
eElement));

                map.put("stat", Utils.getTagValue("stat", eElement));
                map.put("statUpdDt", Utils.getTagValue("statUpdDt",
eElement));

                map.put("lastTsdT", Utils.getTagValue("lastTsdT",
eElement));

                map.put("lastTedt", Utils.getTagValue("lastTedt",
eElement));

                map.put("nowTsdT", Utils.getTagValue("nowTsdT",
eElement));

                System.out.println("busiId : " +
Utils.getTagValue("busiId", eElement));
                System.out.println("statId : " +
Utils.getTagValue("statId", eElement));
                System.out.println("chgerId : " +
Utils.getTagValue("chgerId", eElement));
                System.out.println("stat " + Utils.getTagValue("stat",
eElement));
                System.out.println("statUpdDt " +

```

```

Utils.getTagValue("statUpdDt", eElement));
        System.out.println("lastTsdt " +
Utils.getTagValue("lastTsdt", eElement));
        System.out.println("lastTedt " +
Utils.getTagValue("lastTedt", eElement));
        System.out.println("nowTsdt " +
Utils.getTagValue("nowTsdt", eElement));
        System.out.println("=====");
    }
} catch (Exception e) {
    // TODO: handle exception
}
}

```

응답 메시지 명세 (전략)

항목명(영문)	항목명(국문)	항목 크기	항목 구분	샘플데이터	항목설명
statNm	충전소명	100	1	기후대기관	충전소명
statId	충전소ID	8	1	28260005	충전소ID
chgerId	충전기ID	2	1	02	충전기ID
chgerType	충전기타입	2	1	03	충전기타입 (01:DC차데모, 02: AC완속, 03: DC차데모+AC3상, 04: DC콤보, 05: DC차데모+DC콤보 06: DC차데모+AC3상 +DC콤보, 07: AC3상)
addr	주소	150	1	인천광역시 서구 환경로 42	충전기 주소
location	상세위치	200	0	입구 진입 후 2시 방향	충전기 상세위치

lat	위도	15	1	37.569620	충전기 위도
lng	경도	15	1	126.641973	충전기 경도
useTime	이용가능시간	50	1	24시간 이용가능	충전기 이용가능시간
busild	기관 아이디	2	1	ME	기관 아이디
bnm	기관명	50	1	환경부	기관명
busiNm	운영기관명	50	1	한국자동차환경협회	운영기관명
busiCall	운영기관연락처	20	1	1661-9408	운영기관 연락처
stat	충전기상태	1	1	2	충전기상태 (1: 통신이상, 2: 충전대기, 3: 충전중, 4: 운영중지, 5: 점검중, 9: 상태미확인)
statUpdDt	상태갱신일시	14	1	20210802121020	충전기 상태 변경, 통신이상, 통신복구일시
lastTsdT	마지막 충전시작일시	14	0	20210801121020	마지막 충전시작일시
lastTsdT	마지막 충전종료일시	14	0	20210801123020	마지막 충전종료일시
nowTsdT	충전중 시작일시	14	0	20210802131020	충전중 시작일시
output	충전용량	20	0	50	충전용량 kW (3, 7, 50, 100, 200)
method	충전방식	10	0	단독	충전방식 (단독/동

					시)
zcode	지역코드	2	1	28	시도 코드 (행정구역코드 앞 2자리)
zscore	지역구분 상세 코드	5	0	11680	지역구분 상세(행정구역코드 앞 2자리 + 법정동코드3자리)
kind	충전소 구분 코드	2	0	F0	충전소 구분
kindDetail	충전소 구분 상세코드	4	0	F002	충전소 구분 상세
parkingFree	주차료무료	1	0	Y / N	주차료 (Y: 무료, N: 유료, 값 없는 경우 현장 확인)
note	충전소 안내	200	0	공사로 인해 이용 불가	충전소 이용 안내 사항
limitYn	이용자 제한	1	1	Y / N	이용자 제한 (N:제한 없음)
limitDetail	이용제한 사유	100	0	거주자외 출입제한	이용제한 사유
delYn	삭제 여부	1	1	Y / N	충전기 정보 삭제 여부 (최근 삭제된 충전기 정보 제공)
delDetail	삭제 사유	100	0	고객 요구로 철거	충전기 정보 삭제 사유

실행결과

```

Console Problems Debug Shell Coverage Se
<terminated> DataGoAPI [Java Application] C:\Program File
Result<?xml version="1.0" encoding="
99
===== Item : 1 =====
note:null
statId:ME000001
zcode:11
busiId:ME
lastTsdt:20211028002722
kindDetail:B001
nowTsdt:null
limitYn:N
output:50
chgerId:01
zscore:11110
limitDetail:null
useTime:24시간 이용가능
addr:서울특별시 종로구 종로 157
lat:37.571076
bnm:환경부
stat:9
lng:126.99588
method:단독
kind:B0
busiNm:환경부
parkingFree:Y
delYn:N
statUpdDt:20220316144310
chgerType:03
lastTedt:20211028004608
busiCall:1661-9408
location:null
statNm:종로 공영주차장
delDetail:null
powerType:null

```

Excel 결과 저장 파일

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	
1	station	statId	chgerId	chgerType	addr	location	lat	lng	useTime	statUpdDt	statNm	busiNm	busiNm	busiNm	stat	statUpdDt	lastTsdt	nowTsdt	powerType	output	me
2	종로 공영주차장	ME00000101	03	서울특별시	종로구 종로 157	서울특별시 종로구 종로 157	37.571076	126.99588	24시간 이용가능	20220316	종로 공영주차장	환경부	환경부	환경부	1661-9408	20220316	20211028002722	20211028004608	50	단	
3	세종로 공영주차장	ME00000201	06	서울특별시	종로구 세종로	서울특별시 종로구 세종로	37.573611	126.976011	24시간 이용가능	20210831	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210831	20210826000000	20210826000000	50	단	
4	그랜드캐슬터 서울	ME00000301	06	서울특별시	종로구 그랜드캐슬터	서울특별시 종로구 그랜드캐슬터	37.559352	127.002350	24시간 이용가능	20210910	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210910	20210905000000	20210905000000	50	단	
5	한강전역 공영주차장	ME00000401	03	서울특별시	종로구 한강전역	서울특별시 종로구 한강전역	37.540085	127.002804	24시간 이용가능	20220811	종로 공영주차장	환경부	환경부	환경부	1661-9408	20220811	20220811000000	20220811000000	50	단	
6	마장동사무소 앞(공중전화부스)	ME00000501	06	서울특별시	종로구 마장동	서울특별시 종로구 마장동	37.560995	127.045525	24시간 이용가능	20211025	종로 공영주차장	환경부	환경부	환경부	1661-9408	20211025	20211023000000	20211023000000	50	단	
7	청량리대역역 앞	ME00000701	03	서울특별시	종로구 청량리대역역	서울특별시 종로구 청량리대역역	37.580650	127.047938	24시간 이용가능	20210803	종로 공영주차장	환경부	환경부	환경부	1661-9408	20211025	20211023000000	20211023185509	50	단	
8	이대역 이문점	ME00000801	03	서울특별시	종로구 이대역	서울특별시 종로구 이대역	37.598454	127.061848	10:00-24:00	20211029	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210803	20210714000000	20210714174044	50	단	
9	종로역 앞	ME00001101	03	서울특별시	종로구 종로역	서울특별시 종로구 종로역	37.60208293	127.04002398	10:00-24:00	20220811	종로 공영주차장	환경부	환경부	환경부	1661-9408	20211029	20211026000000	20211026174039	50	단	
10	성북역지하도 공영주차장	ME00001201	06	서울특별시	종로구 성북역	서울특별시 종로구 성북역	37.60936	127.053263	10:00-24:00	20211123	종로 공영주차장	환경부	환경부	환경부	1661-9408	20220811	20220811000000	20220811155552	50	단	
11	부대역 앞	ME00001301	03	서울특별시	종로구 부대역	서울특별시 종로구 부대역	37.625405	127.017897	10:00-24:00	20210906	종로 공영주차장	환경부	환경부	환경부	1661-9408	20211123	20210000000000	20211008142006	50	단	
12	이대역 문정점	ME00001501	03	서울특별시	종로구 이대역	서울특별시 종로구 이대역	37.60047527	126.92047408	10:00-24:00	20210910	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210906	20210821000000	20210821104005	50	단	
13	구리방역 환승주차장	ME00001701	03	서울특별시	종로구 구리방역	서울특별시 종로구 구리방역	37.637910	126.919110	24시간 이용가능	20210903	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210910	20210908000000	20210908140712	50	단	
14	종로역 앞	ME00001801	03	서울특별시	종로구 종로역	서울특별시 종로구 종로역	37.566685	126.898753	08:00-24:00	20210903	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210903	20210903000000	20210903131215	50	단	
15	시암관리공단 대모유수지주차	ME00001901	03	서울특별시	종로구 시암관리공단	서울특별시 종로구 시암관리공단	37.539442	126.943054	24시간 이용가능	20220811	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210910	20210907000000	20210907102642	50	단	
16	종로역 앞	ME00002101	03	서울특별시	종로구 종로역	서울특별시 종로구 종로역	37.52992611	126.87364033	10:00-24:00	20220811	종로 공영주차장	환경부	환경부	환경부	1661-9408	20220811	20220811000000	20220811000000	50	단	
17	해맞이공영주차장	ME00002201	06	서울특별시	종로구 해맞이	서울특별시 종로구 해맞이	37.520062	126.834854	24시간 이용가능	20220811	종로 공영주차장	환경부	환경부	환경부	1661-9408	20220811	20220811000000	20220810211804	50	단	
18	종로역 앞	ME00002301	03	서울특별시	종로구 종로역	서울특별시 종로구 종로역	37.56401856	126.85059572	10:00-24:00	20211007	종로 공영주차장	환경부	환경부	환경부	1661-9408	20211007	20210810000000	20210810162741	50	단	
19	롯데마트 김포공영점	ME00002401	06	서울특별시	종로구 롯데마트	서울특별시 종로구 롯데마트	37.563453	126.803539	24시간 이용가능	20210728	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210728	20210728000000	2021072813035	50	단	
20	메이필드 호텔	ME00002501	06	서울특별시	종로구 메이필드 호텔	서울특별시 종로구 메이필드 호텔	37.546851	126.818613	24시간 이용가능	20210810	종로 공영주차장	환경부	환경부	환경부	1661-9408	20210810	20210810000000	20210810161627	50	단	

CVS파일 형식으로 Kibana 파일 추가

Overview Anomaly Detection Data Frame Analytics **Data Visualizer** Settings

encoded-EvChargerInfo.csv

File contents

First 100 lines

```

1 statId,statId,chargerId,chargerType,addr,location,lat,lng,useTime,busiNm,busiCall,stat,statUpdDt,lastTsdT,lastTsdT,nowTsdT,nowTsdT,powerType,output,method,code,rcode,kind,kindDetail,parkingFree,note,limitYn,limitDetail,delYn,delDetail
2 종로 순환주차장/H000001,01,03,서울특별시 종로구 종로 157,11137.371876,126.95588,24시간 이용가능,배,완전부,완전부,1661-9408,9,20220305144318,20211028002722,20211028004006,,50,단독,11,11130,80,8001,Y,,N,,N
3 세종대 공공주차장/H000002,01,06,서울특별시 용문구 세원로 109,11137.575611,126.370011,24시간 이용가능,배,완전부,완전부,1661-9408,9,20210803124319,20210803023034,20210803094055,,50,단독,11,11130,80,8001,Y,,N,,N
4 그랜드메리디언 서울/H000003,01,06,서울특별시 중구 종로로 287,11137.559352,127.002359,24시간 이용가능,배,완전부,완전부,1661-9408,9,20210910103010,202109051409,2021090521814,,50,단독,11,11140,60,6003,N,,N,,N
5 남산동사거리 순환주차장/H000004,01,03,서울특별시 용문구 남산동 순환로 988,11137.566093,127.040256,24시간 이용가능,배,완전부,완전부,1661-9408,9,20210910095318,20210909095317,20210909095318,,50,단독,11,11170,80,8001,Y,,N,,N
6 남산동사거리 순환주차장/H000005,01,06,서울특별시 용문구 남산동 순환로 214,11137.560659,127.047918,24시간 이용가능,배,완전부,완전부,1661-9408,9,20210225113718,2021021818504,2021021818504,,50,단독,11,11230,60,6002,N,,N,,N
7 장안문화체육공원/H000007,01,03,서울특별시 용문구 장안로 214,11137.560659,127.047918,24시간 이용가능,배,완전부,완전부,1661-9408,9,20210225113718,2021021818504,2021021818504,,50,단독,11,11230,60,6002,N,,N,,N
8 이태원 순환주차장/H000008,01,03,서울특별시 용문구 이태원 순환로 136,11137.558654,127.043484,18:00-24:00,배,완전부,완전부,1661-9408,9,20210911100119,20210714170756,20210714170756,,50,단독,11,11230,60,6002,N,,N,,N
9 홍익대학교 순환주차장/H000011,01,03,서울특별시 용문구 홍익로 76,11137.6030293,127.0400239,18:00-24:00,배,완전부,완전부,1661-9408,9,20210225113718,2021021818504,2021021818504,,50,단독,11,11230,60,6002,N,,N,,N
10 홍익대학교 순환주차장/H000012,01,03,서울특별시 용문구 홍익로 76,11137.6030293,127.0400239,18:00-24:00,배,완전부,완전부,1661-9408,9,20210225113718,2021021818504,2021021818504,,50,단독,11,11230,60,6002,N,,N,,N
11 신촌 순환주차장/H000015,01,03,서울특별시 용문구 신촌 순환로 111,11137.6004757,126.9204740,18:00-24:00,배,완전부,완전부,1661-9408,9,20210909100810,20210803124319,20210803124319,,50,단독,11,11300,60,6002,N,,N,,N
12 신촌 순환주차장/H000016,01,03,서울특별시 용문구 신촌 순환로 111,11137.6004757,126.9204740,18:00-24:00,배,완전부,완전부,1661-9408,9,20210909100810,20210803124319,20210803124319,,50,단독,11,11300,60,6002,N,,N,,N
13 구리역 순환주차장/H000017,01,03,서울특별시 용문구 구리역 순환로 66-30,11137.637910,126.91910,24시간 이용가능,배,완전부,완전부,1661-9408,9,20210910103010,20210908132656,20210908140712,,50,단독,11,11300,80,8003,Y,,N,,N
14 불광역 순환주차장/H000020,01,03,서울특별시 용문구 불광역 순환로 249,11137.566685,126.898735,08:00-24:00,배,완전부,완전부,1661-9408,9,20210909142318,20210909125236,20210909131213,,50,단독,11,11400,60,6002,N,,N,,N

```

Summary

Number of lines analyzed: 100

Format: delimited

Delimiter: ,

Has header row: true

Time field: statUpdDt

Time format: yyyyMMddHHmmss

Override settings Analysis explanation

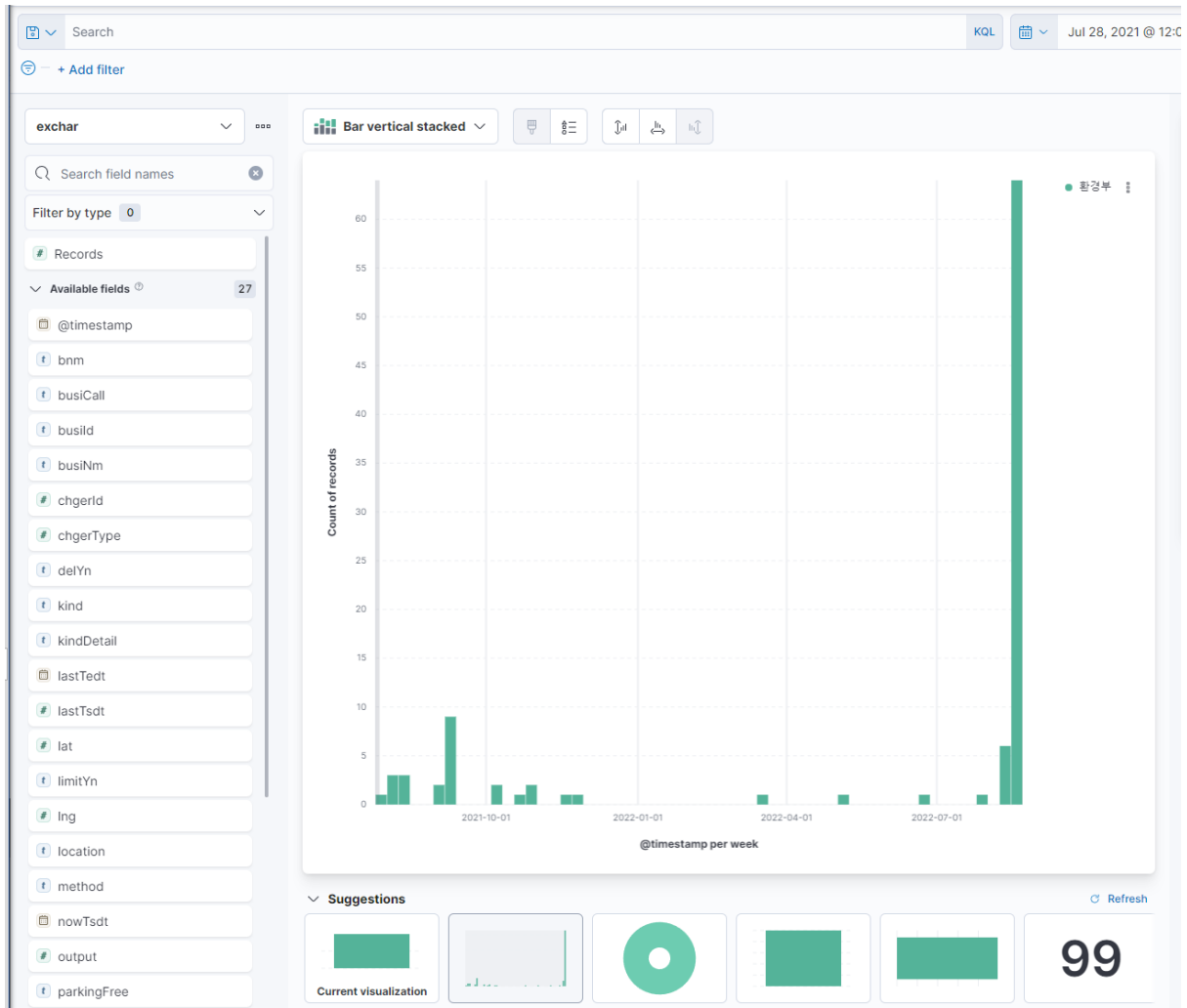
Index를 추가하고

```

/home/hopeki/upload > curl -XPUT "http://20.196.220.19:9200/exchangerinfo?pretty" -d @exchangerinfo_mapping.json -H 'Content-Type:application/json'
{"acknowledged": true,
 "shards acknowledged": true,
 "index": "exchangerinfo"}
/home/hopeki/upload >

```

대시보드를 통해서 데이터를 시각화



(2) 충전기 상태 조회 오퍼레이션 명세

해당 데이터셋을 OPEN API 실험 결과는 다음과 같다 (JAVA 언어)

요청 메시지 명세

항목명(영문)	항목명(국문)	값	항목설명
serviceKey	서비스키	100	인증된 서비스키
pageNo	페이지 번호	1	페이지 번호
numOfRows	한 페이지 결과 수	9999	한 페이지 결과 수 (최소 10, 최대 9999)
period	상태갱신 기간	5	상태갱신 조회 범위(분) (기본값 5, 최소 1, 최대 10)
zcode	지역구분 코드	11 (서울특별시)	시도 코드 (행정구역코드 앞 2자리)

작성 코드

```
String urlGetChargerStatus =
"http://apis.data.go.kr/B552584/EvCharger/getChargerStatus";
    urlBuilder.append("?"+ URLLEncoder.encode("serviceKey","UTF-8")
+ "="+ Utils.getAPIKey()); /*Service Key*/
    urlBuilder.append("&" + URLLEncoder.encode("pageNo","UTF-8") + "=" +
URLLEncoder.encode("1", "UTF-8")); /*페이지 번호*/
    urlBuilder.append("&" + URLLEncoder.encode("numOfRows","UTF-8") + "=" +
URLLEncoder.encode("9999", "UTF-8")); /*한 페이지 결과 수 (최소 10, 최대 9999)*/
    urlBuilder.append("&" + URLLEncoder.encode("period","UTF-8") + "=" +
URLLEncoder.encode("5", "UTF-8")); /*상태갱신 조회 범위(분) (기본값 5, 최소 1, 최대 10)*/
    urlBuilder.append("&" + URLLEncoder.encode("zcode","UTF-8") + "=" +
URLLEncoder.encode("11", "UTF-8")); /*시도 코드 (행정구역코드 앞 2자리)*/
    URL url = new URL(urlBuilder.toString());
    HttpURLConnection conn = (HttpURLConnection) url.openConnection();
    conn.setRequestMethod("GET");
    conn.setRequestProperty("Content-type", "application/json");
    System.out.println("Response code: " + conn.getResponseCode());
    BufferedReader rd;
    if(conn.getResponseCode() >= 200 && conn.getResponseCode() <= 300) {
        rd = new BufferedReader(new
InputStreamReader(conn.getInputStream()));
    } else {
        rd = new BufferedReader(new
InputStreamReader(conn.getErrorStream()));
    }
    StringBuilder sb = new StringBuilder();
    String line;
    while ((line = rd.readLine()) != null) {
        sb.append(line);
    }
    rd.close();
    conn.disconnect();
```

응답 메시지 명세 (전략)

항목명(영문)	항목명(국문)	항목크기	항목구분	샘플데이터	항목설명
busId	기관 아이디	2	1	ME	기관 아이디
statId	충전소ID	8	1	28260005	충전소ID
chgerId	충전기ID	2	1	02	충전기ID
stat	충전기상태	1	1	2	충전기상태 (1: 통신이상, 2: 충전대기, 3: 충전중, 4: 운영중지, 5: 점검중, 9: 상태미확인)
statUpdDt	상태갱신일시	14	1	20190829121020	충전기 상태 변경, 통신이 상, 통신복구 일시
lastTsdT	마지막 충전 시작일시	14	0	20210801121020	마지막 충전시작일시
lastTsdT	마지막 충전 종료일시	14	0	20210801123020	마지막 충전종료일시
nowTsdT	충전중 시작 일시	14	0	20210802131020	충전중 시작일시

실행결과

```

Console Coverage
<terminated> DataGoAPI [Java Application] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe
Response code: 200
81
===== Item : 1 =====
busId : ME
statId : ME174018
chgerId : 01
stat 3
statUpdDt 20220821135057
lastTsdT 20220820223650
lastTsdT 20220820225641
=====
===== Item : 2 =====
busId : ME
statId : ME174025
chgerId : 02
stat 3
statUpdDt 20220821135100
lastTsdT 20220821085916
lastTsdT 20220821094002
-----

```

Excel 파일 저장 결과

	A	B	C	D	E	F	G	H	I	J
1	busild	statld	chgerld	stat	statUpdDt	lastTsdtd	lastTsdtd	nowTsdtd		
2	ME	ME174018	01	2	20220821160038	20220821135056	20220821143056			
3	ME	ME174028	01	3	20220821160037	20220821140153	20220821143223			
4	ME	ME174028	03	3	20220821160030	20220821152351	20220821155300			
5	ME	ME174041	02	3	20220821160247	20220821152439	20220821154357	20220821160246		
6	ME	ME183007	01	3	20220821160306	20220821101428	20220821105732	20220821160304		
7	ME	ME18B166	01	2	20220821155913	20220821154801	20220821155912			
8	ME	ME18B170	01	2	20220821160301	20220821154514	20220821160259			
9	ME	ME18B280	12	3	20220821160210	20220821151323	20220821155626	20220821160210		
10	ME	ME19A244	21	2	20220821155944	20220821082212	20220821090218			
11	ME	ME19A601	24	2	20220821160115	20220821152110	20220821160115			
12	ME	ME19A801	01	3	20220821160146	20220821150633	20220821153407	20220821160146		
13	ME	ME19Y284	22	3	20220821160020	20220821132507	20220821134854	20220821160020		
14	ME	ME20A410	03	2	20220821160050	20220821152042	20220821160042			
15	EP	EP210012	01	2	20220821155904	20220821100812	20220821120840	20220821134858		
16	EP	EP220136	14	3	20220821160029	20220820100852	20220821120840	20220821160028		
17	EV	EV000604	01	2	20220821160212	20220821135836	20220821154121	20220821135836		
18	EV	EV000655	05	3	20220821155913	20220819074313	20220819165210	20220821155834		
19	EV	EV001459	02	2	20220821155912	20220821133608	20220821153944	20220821133608		
20	EV	EV001574	03	3	20220821160208	20220821091957	20220821131529	20220821160158		
21	EV	EV002387	02	2	20220821160312	20220821113345	20220821154322	20220821113345		

나) 한국전력공사_전기차 충전소 운영정보

기본정보

데이터 형태	오픈 API	제공기관	한국전력공사
분류체계	일반공공행정 - 일반행정	데이터포맷	XML
API 유형	REST	키워드	전기차충전소 등
등록일자	2019-06-26	수정	2021-04-27
비용부과유무	무료	신청가능 트랙픽	
심의유형	개발단계: 자동승인 /운영단계: 자동승인		
이용허락범위	이용허락범위 제한 없음		

데이터셋을 활용할 때 위에 같은 기본정보가 확인할 필요가 있다. 이외에 추가로 Docx 파일 형태도 여러 정보 (서비스 명세, API 실행예제)를 확인할 수 있다. 특히 개발할 때 API를 제대로 호출하기 위해서 꼭 Parametar같은 전송 데이터를 정상적으로 체크해야하고 또는 리턴정보도 제대로 출력되었는 확인해야 한다. [그림9]은 API를 호출할 때 전송 필요한 Paramater 및 응답 메시지를 어떻게 출력되는지 보여주고 있다.

그림9. API 호출 예시

① 요청 메시지 명세

항목명(영문)	항목명(국문)	항목크기	항목구분	샘플데이터	항목설명
ServiceKey	서비스키	400	1	공공데이터포털에서 받은 인증키	공공데이터포털에서 받은 인증키
numOfRows	한페이지결과수	4	1	10	한 페이지 결과 수
pageNo	페이지 번호	4	1	1	페이지 번호
addr	충전소 주소	255	0	전라남도 나주시 전력로 55	검색대상 충전소주소

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

② 응답 메시지 명세

항목명(영문)	항목명(국문)	항목크기	항목구분	샘플데이터	항목설명
resultCode	결과코드	2	1		결과코드
resultMsg	결과메세지	50	1		결과메세지
numOfRows	한 페이지 결과 수	4	1	10	한 페이지 결과 수
pageNo	페이지 번호	4	1	1	페이지 번호
totalCount	전체 결과 수	4	1	8	전체 결과 수
items	목록		0..n		목록
addr	충전소주소	255	1	전라남도 나주시 전력로 55	충전소주소
chargeTp	충전기타입	1	1	2	1 : 완속 2 : 급속
cpId	충전기ID	12	1	13	충전기ID
cpNm	충전기명칭	20	1	급속01	충전기명칭
cpStat	충전기 상태 코드	2	1	1	1 : 충전가능 2 : 충전중 3 : 고장/점검 4 : 통신장애 5 : 통신미연결
cpTp	충전방식	2	1	8	1 : B타입(5핀) 2 : C타입(5핀) 3 : BC타입(5핀)

3.2. 유기견보호: 농림축산식품부 농림축산검역본부 동물보호관리시스템 유기동물 조회 서비스 동물보호관리시스템의 유기동물 정보를 조회할 수 있다.

- 농림축산식품부 농림축산검역본부_동물보호관리시스템 유기동물 정보 조회 서비스 ([링크](#))

개요

유기동물주인의 실수, 혹은 의도적인 목적으로 인하여 버려진 이국적인 동물, 혹은 반려동물을 뜻한다. 유기동물들은 주로 주인의 죽음이나, 혹은 동물들이 너무 커지거나 질병에 걸린 경우 발생한다. 한국일보에 따르면 지난해 유기동물(유실동물 포함)이 약 12만 마리 발생한 것으로 분석됐다. 유기동물 발생 수는 2020년 대비 소폭 감소했지만, 1세 미만과 비품종견 비율은 더욱 상승했다. 이를 해결하기 위해서 여러 작업을 진행해야 되지만 제공한 유기동물 데이터셋을 통해서 정보를 조회할 수 있게 하는 방법이 하나라고 생각한다. 그래서 해당 데이터 셋을 분석할 만할거라고 생각한다.

데이터 셋

해당 데이터셋 기본정보는 다음과 같다.

데이터 형태	오픈 API	제공기관	농림축산식품부 농림축산검역본부
분류체계	동물보호과	데이터포맷	XML – JSON
API 유형	REST	키워드	유기동물
등록일자	2022-02-11	수정	2022-02-11
비용부과유무	무료	신청가능 트랙픽	10,000
심의유형	개발단계: 자동승인 /운영단계: 자동승인		
이용허락범위	이용허락범위 제한 없음		

해당 데이터셋에서도 제공된 Docs 파일을 다운로드하면 데이터 셋에 대한 여러 정보를 확인할 수 있고 API를 호출되는 예제도 제공한다. 이를 위에 기재되어 있는 링크를 접속한 다음에 Docs 파일을 쉽게 다운로드할 수 있다.

해당 상세기능 목록은 다음과 같다.

번호	API명(국문)	상세기능명(영문)	상세기능명(국문)
1	동물보호관리시스템 유기 동물 조회 서비스	sido	시도 조회
2		sigungu	시군구 조회
3		shelter	보호소 조회
4		kind	품종 조회
5		abandonmentPublic	유기동물 조회

이에 따라서 Open API를 호출할 때 필요한 데이터셋을 확인해서 URL을 수정 및 Paramater 값이 조절하면 된다. 본 과제에서 시도 조회 API를 Java 구현해 봤다.

출력 결과

```
===== Item : 1 =====
```

```
orgCd:6110000
```

```
orgdownNm:서울특별시
```

```
===== Item : 2 =====
```

```
orgCd:6260000
```

```
orgdownNm:부산광역시
```

```
===== Item : 3 =====
```

```
orgCd:6270000
```

```
orgdownNm:대구광역시
```

```
===== Item : 4 =====
```

```
orgCd:6280000
```

```
orgdownNm:인천광역시
```

```
===== Item : 5 =====
```

```
orgCd:6290000
```

```
orgdownNm:광주광역시
```

	A	B	
1	orgCd	orgdownNm	
2	6110000	서울특별시	
3	6260000	부산광역시	
4	6270000	대구광역시	
5	6280000	인천광역시	
6	6290000	광주광역시	
7	5690000	세종특별자치시	
8	6300000	대전광역시	
9	6310000	울산광역시	
10	6410000	경기도	
11	6420000	강원도	
12	6430000	충청북도	
13	6440000	충청남도	
14	6450000	전라북도	
15	6460000	전라남도	
16	6470000	경상북도	
17	6480000	경상남도	

3.3 한우 낙농 · 축산물품질평가원 축산물경락가격정보 : 축산물 소 · 돼지 · 닭 · 계란 · 오리 에 대한 등급판정 결과 고기 및 부산물 정보 도매시장 정보 경매 현황 도폐 육질.

➤ [축산물품질평가원_축산물경락가격정보\(링크\)](#)

개요

최근에 건강에 관심사가 많아지면서 평소에 먹던 음식물에 대한 평가 주목을 받게 된다. 현재 축산물 품질평가원에서는 크게 세가지로 분류되는 정보 를 제공하고 있는데, 첫째는 축산물이력정보, 둘째, 축산물 유통정보, 셋째 축산물등급판정확인서 발급정보 로 볼 수 있다. 이를 모두 축산관련 기관 및 업계에서는 업 무처리를 위한 필수 정보로 많이 사용된다. 위에 축산물품질평가원에서 제공하고 있는 데이터셋을 통해서 여러 가지 정보를 확인할 수 있으면 평소에 먹던 음식물을 더 이해할 수 있는것으로 판단한다.

데이터 셋

해당 데이터셋 기본정보는 다음과 같다.

데이터 형태	오픈 API	제공기관	축산물품질평가원
분류체계	농림 - 임업·산촌	데이터포맷	XML
API 유형	REST	키워드	축산물,등급,소·돼지·

			닭·계란·오리
등록일자	2018-07-13	수정	2021-10-28
비용부과유무	무료	신청가능 트랙픽	1000
심의유형	개발단계: 자동승인 /운영단계: 자동승인		
이용허락범위	이용허락범위 제한 없음		

해당 데이터셋에서도 제공된 Docs 파일을 다운로드하면 데이터 셋에 대한 여러 정보를 확인할 수 있고 API를 호출되는 예제도 제공한다. 이를 위에 기재되어 있는 링크를 접속한 다음에 Docs 파일을 쉽게 다운로드할 수 있다.

해당 데이터는 나. 오퍼레이션 목록은 다음과 같다. 본 과제에서 제주 돼지도체 등급별 경락가격 정보 조회 API를 Java 구현해 봤다.

일련 번호	서비스명(국문)	상세기능명(영문)	상세기능명(국문)
1	축산물경락가	auct/pigJeuGrade	제주 돼지도체 등급별 경락가격 정보 조회
2	격 조회 서비스	auct/beefGrade	쇠고기 부분육 경락가격 조회

3. 4. 농경락: '농경락'은 농산물의 실시간 경매 가격 정보 도매시장 도매법인 산지별 정보를 제공.

- 농림수산물교육문화정보원_농수축산물 도매시장 상세 경락가격 ([링크](#))

개요

음식물중에서 가장 많이 소요되는 항목이 반드시 농산물이다. 이에 따라서 전국 도매시장에서 거래되어 있는 농산물들이 분석할 수 있으면 농식품 소비나 농산물을 구매하는데 많이 도움이 될거라고 판단한다.

데이터 셋

해당 데이터셋 기본정보는 다음과 같다.

데이터 형태	오픈 API	제공기관	농림수산물교육문화 정보원
분류체계	농림 - 임업·산촌	데이터포맷	JSON+XML
API 유형	REST	키워드	경락가격
등록일자	2016-01-06	수정	2020-10-19
비용부과유무	무료	신청가능 트랙픽	1000
심의유형	개발단계: 자동승인 /운영단계: 자동승인		

이용허락범위	이용허락범위 제한 없음
--------	--------------

해당 데이터셋에서도 제공된 Docs 파일을 다운로드하면 데이터 셋에 대한 여러 정보를 확인할 수 있고 API를 호출되는 예제도 제공한다. 이를 위에 기재되어 있는 링크를 접속한 다음에 Docs 파일을 쉽게 다운로드할 수 있다.

해당 데이터는 나. 오퍼레이션 목록은 다음과 같다. 본 과제에서 원천정산경락가격품목목록 조회 API를 Java 구현해 봤다.

일련 번호	서비스명(국 문)	오퍼레이션명(영문)	오퍼레이션명(국문)	메시지명(영 문)
1	농수축산물 도매시장 상 세 경락가격 서비스	getExactProdPriceList	원천정산경락가격품목 목록조회	N/A
2		getExactMarketPriceList	원천정산경락가격도매시 장목록조회	N/A
3		getRealProdPriceList	원천실시간경락가격품목 목록조회	N/A
4		getRealMarketPriceList	원천실시간경락가격도매 시장목록조회	N/A

실행결과


```

===== Item : 5 =====
aucSeCode:2
aucSeNm:정가수의
catgoryCode:10
catgoryNewCode:10
catgoryNewNm:엽경채류
catgoryNm:엽경채류
cprInsttCode:22000111
cprInsttNewCode:6041404
cprMtcCode:536000
cprMtcNm:전라남도 해남군
cprUsePrdlstCode:100100
cprUsePrdlstNm:배추(일반)
delngDe:20151202
delngPrut:7
delngQy:200
insttNewNm:청과상장예외
ledgNo:15112903007
rnum:5
sbidPric:4000
shipmntSeCode:3
shipmntSeNm:개별
sleSeqn:1000
stdFrmlcNewCode:1ZZ
stdFrmlcNewNm:기타
stdMgNewCode:1ZZ
stdMgNewNm:기타
stdMtcNewCode:59000
stdMtcNewNm:전라남도 해남군
stdPrdlstCode:1001
stdPrdlstNewCode:1001
stdPrdlstNewNm:배추
stdPrdlstNm:배추
stdQlityNewCode:13
stdQlityNewNm:보통
stdSpciesCode:100101
stdSpciesNewCode:100101
stdSpciesNm:봄배추
stdUnitNewCode:12
stdUnitNewNm:kg
whsalMrktCode:220001
whsalMrktNewCode:1041401
whsalMrktNewNm:대구북부도매시장
whsalMrktNm:대구북부도매시장

```

Excel 파일 저장

	A	B	C	D	E	F	G	H	I	J	
1	aucSeCode	aucSeNm	catgoryCode	catgoryNewCode	catgoryNewNm	catgoryNm	cprInsttCode	cprInsttNewCode	cprMtcCode	cprMtcNm	cp
2		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
3		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
4		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
5		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
6		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
7		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
8		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
9		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
10		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
11		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
12		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
13		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
14		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
15		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	
16		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
17		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
18		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
19		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
20		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
21		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
22		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
23		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
24		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
25		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
26		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
27		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
28		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
29		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
30		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
31		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	210000	강원도 강릉시	
32		2 정가수의	10	10	업경채류	업경채류	22000111	6041404	536000	전라남도 해남군	

4. 가점사항

4.1 공공데이터의 API 를 이용하여 동적으로 Logstash 연동

Logstash에서 REST API를 연동하기 위해서 방법이 크게 2가지 있다. 하나는 Logstash에서 제공하는 http_poller플러그인을 통해서 진행하는 것이다. 하나는 REST API 호출 결과를 데이터베이스 (Mysql)에서 저장해서 jdbc 플러그인으로 가져와서 전송할 수 있다. 본 과제에서 시간상으로 해당항목을 진행 못하고 향후에 진행할 예정이다.

4.2 지정과제에서 명시된 외의 데이터를 참조하여 시각화

본 과제에서 명시된 외의 데이터가 아니라 “한국연구재단_외국교육기관 및 외국인학교 종합안내”를 찾아서 실험해봤다. 해당 데이터를 사용하면 외국인에 대한 수많은 정보를 확인할 수 있으면 API가 더 여러가지 제공하고 있다.

해당 데이터 기본정보는 다음과 같다.

데이터 형태	오픈 API	제공기관	한국연구재단
분류체계	교육 - 고등교육	데이터포맷	XML
API 유형	REST	키워드	외국교육기관
등록일자	2021-08-31	수정	2021-10-19

비용부과유무	무료	신청가능 트랙픽	1000
심의유형	개발단계: 자동승인 /운영단계: 자동승인		
이용허락범위	이용허락범위 제한 없음		

출력 결과

```
▼<item>
  <siBasId>1026</siBasId>
  <schoolId>1</schoolId>
  <basYear>2020</basYear>
  <schoolNmK>한국외국인학교(서울캠퍼스)</schoolNmK>
  <schoolNmE>Korea International School - Seoul</schoolNmE>
  <sidCd>11</sidCd>
  <addrK>서울특별시 강남구 개포로 408</addrK>
  <addrE>408 Gaepo-ro,Gangnam-gu, Seoul, 06324, Korea</addrE>
  <openCourse>유치원,초등학교|Kindergarten,Primary</openCourse>
  <useLanguage>영어|English</useLanguage>
  <schoolType>ST01</schoolType>
  <tel>82-2-3496-0509</tel>
  <homepage>http://kisseoul.org/</homepage>
  <status>1</status>
  <mainImage>20181115113923_(200x155)KJM_8887.jpg</mainImage>
  <regDt>2018-11-15 11:27:17.0</regDt>
  <openYmd>1999-08</openYmd>
  <studyStdYmdK>8월 둘째주</studyStdYmdK>
  <studyStdYmdE>Mid-August</studyStdYmdE>
  <schoolType1>Y</schoolType1>
  <schoolType2/>
  <schoolType3/>
  <useYn>Y</useYn>
  <adminNmK/>
  <adminNmE/>
  <adminPostK/>
  <adminPostE/>
  <adminTel>02-3496-0520</adminTel>
  <adminEmail/>
  <eduOfficeUrl/>
  <eduOfficeTel/>
  <schoolType4/>
  <schKor/>
  <schProf/>
  <zipCode1>06324</zipCode1>
  <zipCode2/>
  <addrDetailK/>
  <addrDetailE/>
  <schoolType5/>
  <schMngCorp/>
  <quarterGbn/>
  <quarterDate/>
</item>
</items>
```

4.3 MySQL 같은 RDB 연동로 Logstash 연동

본항목에서 해당 작업을 진행하기 위해서 추가로 MySQL를 설치하는 작업을 진행해야 한다.

```
# package update
Sudo apt update
```

```
# mysql install
sudo apt install mysql-server

# start the server
sudo systemctl start mysql.service

# 설치한 다음에 User를 추가 권한
mysql -u root -p
create user '사용자'@'localhost' identified by '비밀번호';
grant all privileges on *.* to '사용자'@'localhost';
grant all privileges on DB이름.* to '사용자'@'localhost';
```

Mysql를 설치 완료한 다음에 데이터 베이스를 입력해야 한다. (*.sql 파일 추가하거나 Mysql Workbench에서 UI로 추가함. 데이터 베이스를 입력완료한 다음에 Logstash를 동기화를 진행한다. 동기화를 진행하기 위해서 위에 생성된 mysql User 정보를 입력하고 Query를 원하는 추가하면 된다. 추가로 필터가 필요입력도 가능하고 Elasticsearch로 output로 설정할 수 있다. (주의: mysql user 확인 및 Elasticsearch 출력되는 Index)

```
input{
  jdbc{
    jdbc_driver_library => "/home/hopekr/upload/mysql-connector-java-8.0.30.jar"
    jdbc_driver_class => "com.mysql.cj.jdbc.Driver"
    jdbc_connection_string => "jdbc:mysql://localhost:3306/elastic?characterEncoding=UTF-8&serverTimezone=UTC"
    jdbc_user => "huong"
    jdbc_password => "huong1997"
    statement => "SELECT * from plas_problems"
  }
}

filter{
  mutate{
    add_field => { "[location][lat]" => [ "%{lat}" ] }
    add_field => { "[location][lon]" => [ "%{lon}" ] }
    convert => [ "[location]", "float" ]
    remove_field => [ "%{lat}", "%{lon}" ]
  }
}

output {
  elasticsearch {
    hosts => ["http://20.196.220.19:9200"]
    index => "air"
    #user => "elastic"
    #password => "changeme"
  }
}
```

4.4 Crawling 같은 제의 3방법을 통해서 데이터 취득

본 항목에서 진행하기 위해서 우선 Crawling 또는 Web Crawling를 어떤기술인지 이해가 필요하다. 쉽게 정의하자고 하면 Web Crawling란 웹 사이트에서 원하는 정보를 추출하는 것을 의미한다. 즉 웹이 기본적으로 HTML형태(어떤 정형화된 형태)로 되어 있기 때문에, 이런 규칙을 분석해서 우리가 원하는 정보들만 뽑아오는 것이다. Web Crawling는 일반적인 메커니즘은 4단계가 있으며 대상선정, 데이터 로드, 데이터 분석, 데이터 수집이다. 기술 개발자는 작성하는 언어에 따라서 사용하는 라이브러리가 다를 수 있고 로드 및 분석 단계도 다를 수 있다. 본 과제에서 단순한 대상을 하나 선정해서 크롤링을 진행과정은 다

음과 같다.

1. 대상 선정

웹 사이트는 고유한 ID를 가지며 URL라고 부른다. (www.google.com 등). 최근에 웹개발자가 필요하는 회사가 많아지면서 Sarami 사이트를 통해서 “웹개발자”키워드로 검색해서 대상을 선정한다.

URL: https://www.saramin.co.kr/zf_user/search?searchword=웹개발자

2. 데이터 로드

본 과제에서 Java언어로 단순히 크롤링 함수를 만들어서 코드 및 실행 결과는 다음과 같다. 위에 선정된 링크에서 접속되면 조회된 결과를 Crawling를 통해서 출력된것으로 확인되었다.

```
public static void main(String[] args) throws IOException {

    //Set the URL to connect
    String URL =
    "https://www.saramin.co.kr/zf_user/search?searchword=웹개발자";

    //Fetch and parse HTML file from the web into a Document object
    Document docs = Jsoup.connect(URL).timeout(6000).get();
    Elements contents = docs.select("div.content");

    for (Element element : contents.select("div.item_recruit")) {
        System.out.println(contents);
    }

}
```

```
<div class="item_recruit" value="43712017" data-data_layer="keyword_free|paid_n_quick">
<div class="area_job">
<h2 class="job_title"><a target="_blank" title="웹개발자 (JAVA) 정규직 신입 모집" onclick="try{s_trackApply(this, 'search', 'generic')}catch(e){};" rel="" class="data_layer" data-data_layer="keyword_free|paid_n" href="/zf_user/jobs/relay/view?vi
<div class="tooltipwrap wrap_scrap"><a style="cursor:pointer" class="icon_scrap_star" onclick="saramin.btnJob('scrap',this,'','list');" scrapped="n" rec_id="43712017" data-pattern="{&quot;t_category&quot;:&quot;&quot;search&quot;,&quot;t_cont
<div class="tooltip sri_tooltip_scrap"><span class="tail_tail_bottom_center"></span>
<div class="tooltipCont txtCenter">
스crap
</div>
</div>
<div class="job_date"><span class="date">-- 08/31(수)</span> <button class="sri_btn_xs" title="클릭하면 임시저장할 수 있는 정보입니다." onclick="try{quickApplyForm('43712017','','t_category=search&amp;t_content=generic&amp;t_scid=&amp;search_uid=
</div>
<div class="job_condition"><span><a target="_blank" href="/zf_user/area-recruit/area-list/area/101000/areamode/mcode">사용</a> <a target="_blank" href="/zf_user/area-recruit/area-list/area/101000">공인구</a></span> <span>공민구</span></div>
</div>
<div class="job_sector"><a target="_blank" href="/zf_user/jobs/list/job-category?cat_kewd=84">역전도/서버개발</a>, <a target="_blank" href="/zf_user/jobs/list/job-category?cat_kewd=87">원격일</a>, <a target="_blank" href="/zf_user/jobs/list/
</div>
</div>
<div class="area_corp"><strong class="corp_name"> <a href="/zf_user/company-info/view?csn=TfpQ70ZvME3vRUs3Y1JC52FKV85ud209" target="_blank" class="track_event data_layer" data-track_event="total_search|search_recruit|com_info_btn" data-d
<div class="tooltipwrap wrap_interested_corp">
<button type="button" csn="TfpQ70ZvME3vRUs3Y1JC52FKV85ud209" title="관심기업 등록" del_fl="n" class="interested_corp" onclick="try{saramin.btnJob('favor', this, '', 'list')}catch(e){}" onmouseover="Saramin.favorTooltip(this, 'on');" onmou
<div class="tooltip">
<span class="tail_tail_bottom_center"></span>
<div class="tooltipCont txtCenter">
관심기업 등록
</div>
</div>
</div>
</div>
```

통합검색

채용정보

기업정보

'웹개발자'에 대한 취업정보

채용정보 총 16,911건

☒ 일반 채용정보
 ☒ 파견/대행
 ☒ 헤드헌팅

☒ 입사지원 만 보기
 ☒ 조회한 공고 제외
 ☒ 취업촉하금만 보기

기업형태 전체 관련도순 40개씩

(주)원드롭 모바일/웹 서비스, AI 개발 모집(전문연구요원가능) ☆ ~ 08/29(월) 채용 홈페이지
 경기 성남시 중원구 경력무관 고졸↑ 정규직·전문연구요원
 웹개발, 머신러닝, AI(인공지능), iOS, 유지보수 외 | 수정일 22/07/27

(주)원드롭
 기업정보

[서울형 강소기업] Java 기반 웹개발자 채용(신입/경력) ☆ ~ 08/22(월) 입사지원
 서울 영등포구 경력무관 대졸↑ 정규직·계약직
 웹개발, 임베디드, Java, GIS, S-시스템통합 외 | 등록일 22/08/17

(주)내가시스템
 기업정보

웹프로그래머 모집(JAVA 개발자)-신입 및 경력 ☆ ~ 08/22(월) 입사지원
 서울 성동구 신입·경력 초대졸↑ 정규직
 웹개발, HTML, Java, 솔루션, HTML5 외 | 등록일 22/08/16

(주)커머스웨어
 기업정보

[개발팀] PHP 웹개발자 모집 채용 공고(경력직) ☆ ~ 08/29(월) 입사지원
 대구 중구 경력 2~10년 학력무관 정규직·계약직
 웹개발, Java, PHP, 소프트웨어개발 | 수정일 22/08/03

다담소프트(주)
 기업정보

2022년도 웹개발자 신입/경력(대전,광주) 채용 ☆ 채용시 입사지원
 (주)아카넷

3. 데이터 분석

가지고 온 데이터는 “웹개발자” 키워드로 검색된 결과이며 현재 웹 웹개발자를 구하는 회사들이 출력된 것으로 확인된다. 검색페이지에서만 봐서 단순한 정보만 출력되어 있으며 회사명, 위치, 분야, 직원 타입 등이다. 다만, 이를 봐도 웹개발 분야중에서 회사들이 어떤 기술이 개발자 많이 필요하는지 판단할 수 있고 신입-경력이 많이 필요하는지 판단할 수 있다.

4. 데이터 수집

위에 분석된 데이터를 수집하기 위해서 HTML에서 작성된 Class나 Id를 찾아야 한다. 이를 찾을 다음에 jsoup 패키지에서 구현된 Document클래스를 통해서 수집할 수 있다. 데이터를 수집한 다음에 Excel파일 같은 형태를 저장하면서 더 깊게 분석할 수 있다. 본 과제에서는 시간상으로 해당 작업을 향후에 진행할 예정이다.

4.5 지도(Map) 기반 데이터 연동

참고 문서)

1. <https://www.data.go.kr/>