데이터베이스 최종 보고서

제목: **COVID-19 알림이**

제출일:   2021년 12월 08일

1반 9조

팀 원: 201820802 강선규

201823780 정석화

201720738 주정호

201621886 김영희

**목차**

[1. Introduction 3](#_Toc90234087)

[1.1 개발 목표 3](#_Toc90234088)

[1.2 주제선정 배경 및 기대 효과 3](#_Toc90234089)

[2. Requirements 4](#_Toc90234090)

[2.1 주요 기능 5](#_Toc90234091)

[2.2 Description of Mini world 6](#_Toc90234093)

[3. Data Schema 6](#_Toc90234094)

[3.2 ER Diagram 7](#_Toc90234096)

[3.3 Data Table 8](#_Toc90234097)

[4. DB Implementation 12](#_Toc90234098)

[4.1 DB Schema 12](#_Toc90234099)

[4.2 Constraints 13](#_Toc90234100)

[4.3 Case scenarios for Requirements 19](#_Toc90234101)

[4.4 QUERY PROCESSING OPTIMIZATION 21](#_Toc90234102)

[5. Result 22](#_Toc90234103)

[5.1 Relational model 22](#_Toc90234104)

[5.2 WEB Application 26](#_Toc90234105)

[6. Crawling Data & DevOps 31](#_Toc90234106)

[6.1 Crawling Data 31](#_Toc90234107)

[6.2 DevOps 36](#_Toc90234108)

[7. 참고자료 37](#_Toc90234109)

## **1. Introduction**

# **1.1 개발 목표**

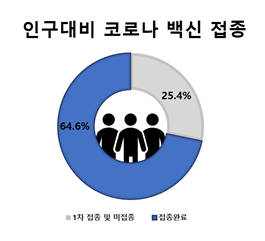
코로나19 백신 예방 접종이 전연령층을 대상으로 진행되고 있다. 본 프로젝트는 코로나 확진 정보에 백신 접종 정보를 추가해 통합 데이터베이스를 구축하는 것을 목표로 한다. 따라서 지역별, 날짜별로 사용자가 빠르게 정보를 확인할 수 있도록 돕는다.

# **1.2 주제선정 배경 및 기대 효과**

* **주제선정 배경**
* **백신접종률 증가에 따른 백신 접종에 관한 관심증대**

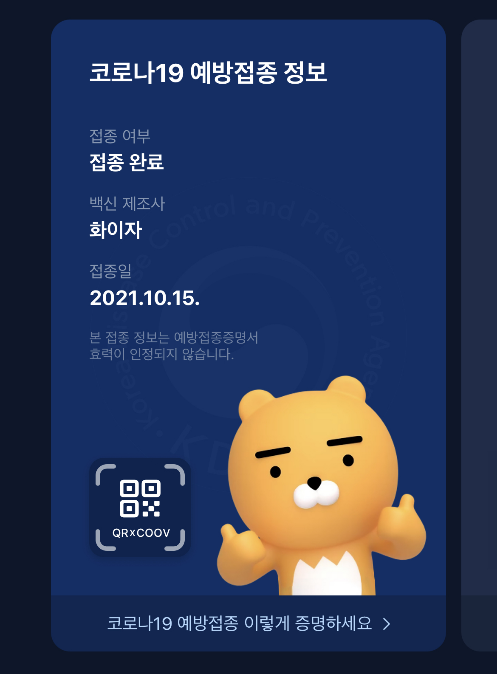
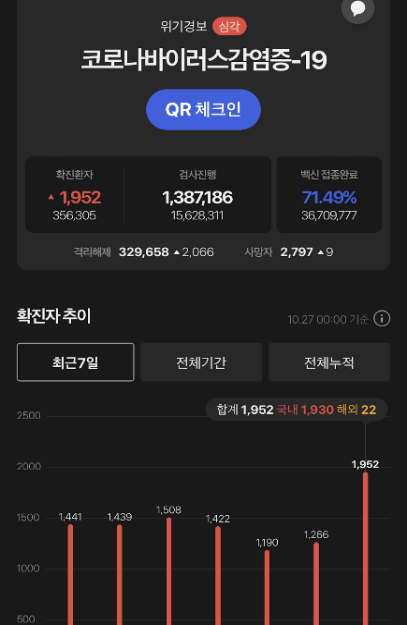
코로나 예방 백신 접종률이 10월 14일 기준 인구대비 60%를 넘어서면서 2019년부터 시작된 코로나19 사태가 새로운 ‘with코로나’시대를 맞이하고 있다.

백신 접종에 대한 관심과 참여로 접종완료율은 먼저 접종을 시작한 프랑스 67.4%, 영국 66.7%, 이스라엘 65.0% 등의 국가들과 유사하다. 또한 백신관련 정보를 전달하고 있는 국민 비서 ‘구삐’의 가입자는 9월이후 1,300만명을 넘어섰다. 백신 접종률 증가와 “with코로나”로 인한 방역 정책변화로 코로나 및 백신에 대한 관심은 높아지고 있다.



* **코로나19 유사 앱**

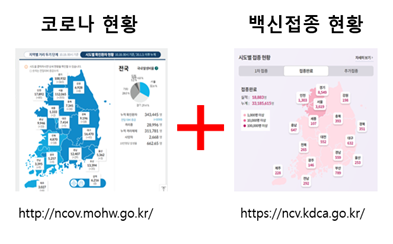
기존의 코로나19 관련 알림 앱들은 백신 종류별 접종자 수까지 세세하게 제공하고 있지는 않았다.  본 프로젝트에서는 상세한 백신 접종 정보를 날짜별로 시각화할 예정이다.

* **기대효과**

**분리된 정보로 인한 불편함 해결**

기존 코로나 사이트는 백신별 현황, 지역별 등 여러가지 데이터가 분리된 상태로 한 눈에 데이터를 파악하기 어렵다. 본 프로젝트는 분리된 데이터들을 크롤링하고 취합하여 데이터베이스에 저장하고 관리할 것이다. 따라서 종합된 데이터를 사용자에게 시각적으로 제공할 수 있다. 나아가서 사용자가 백신 접종 상태 및 신상정보를 등록함으로써 백신 접종 관리의 편리함을 높인다.



2. Requirements

# **2.1 주요 기능**

* **사용자**

1) 지역별 확진 및 접종 현황 조회

지역별 코로나 확진자 수, 백신 접종자 수, 거리두기 단계를 조회할 수 있다.

2) 날짜별 접종 정보 조회

확진 현황 및 날짜별 백신 종류별 접종자 수를 조회할 수 있다.

3) 사용자 백신 정보 등록

사용자의 신상 정보 및 백신 접종 상태를 등록할 수 있다.

4) 지역구 기준 다양한 정보 확인

사용자의 지역(시), 자치구(구)를 기준으로 코로나 확진, 선별진료소를 알 수 있다.

* **개발자**

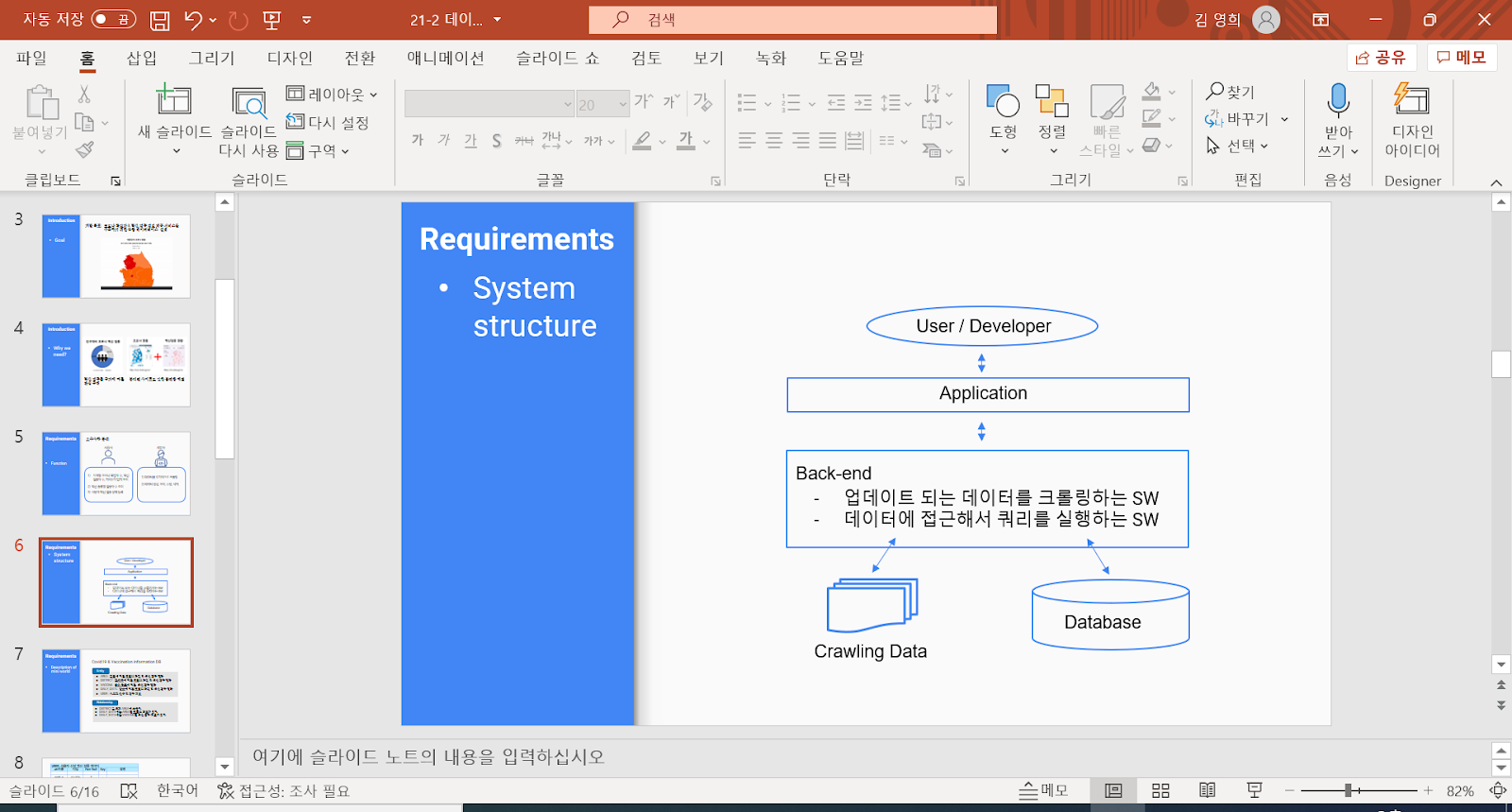
1) 데이터 크롤링

데이터를 주기적으로 크롤링할 수 있다.

2) 데이터 관리

데이터 생성, 조회, 삭제할 수 있다.

* **SYSTEM STRUCTURE**



COVID-19알리미의 통합 데이터베이스의 액터는 사용자와 개발자 총 2명이다. 사용자는 어플리케이션을 통해 [지역별 조회, 날짜별 조회, 사용자 정보 등록, 사용자 지역구 기준 조회] 기능을 수행할 수 있다. 개발자는 애플리케이션 서버를 통해 [데이터 크롤링, 데이터 관리] 기능을 수행할 수 있다. 데이터베이스는 MySQL을 사용하고, 참조 사이트에서 데이터를 수집한다.

# **2.3 Description of Mini world**

* Entity

AREA : 지역에 따른 코로나 확진 및 백신 접종 현황

DISTRICT : 자치구에 따른 코로나 확진 및 백신 접종 현황

VACCINE : 백신 종류에 따른  백신 접종 현황

DAILY\_DATA : 날짜에 따른 코로나 확진 및 백신 접종 현황

SOCIAL\_DIST : 거리두기 단계에 따른 내용

USER : 사용자 신상 및 접종 정보

TODAY\_AREA : 당일 해당 지역의 확진자 수

HOSPITAL : 지역별 코로나 선별진료소

* Relationship

DISTRICT와 SOCIAL\_DIST는 특정 AREA에 속한다.

DAILY\_DATA에는 AREA별 코로나 정보가 있다.

DAILY\_DATA에는 VACCINE별 백신 접종 정보가 있다.

TODAY\_AREA에는 USER의 지역의 실시간 코로나 감염 현황 정보가 있다.

# **3. Data Schema**

# **3.1 수정 사항**

본 프로젝트는 초기에 USER entity와 다른 entity사이의 관계성 부족과, 기존 서비스와의 차별성이 부족하다는 피드백을 받았다.

기존 E-R Model에 relation과 entity를 추가하는 방향으로 DB 스키마 디자인을 개선했다. sql query를 추가하고, 새로운 데이터를 크롤링해서 사용자의 지역구를 중심으로 다양한 정보를 제공한다.

따라서 기존의 entity간의 약한 관계성 보완하고, 본 프로젝트의 차별화된 점을 보여줄 수 있을 것으로 기대한다. 수정 사항 :

1. Entity & relation 추가 :

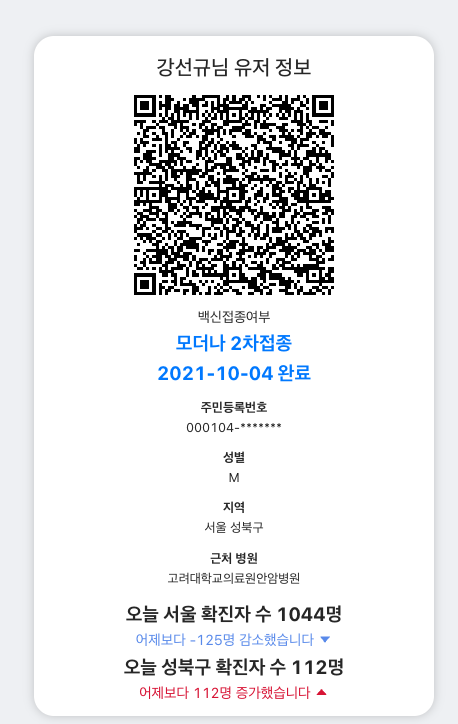
TODAY\_AREA 테이블을 추가해서 today\_area, area entity들과 user entity사이의 연결관계를 만들었다. [수정 후 E-R Diagram] 참조

1. SQL 쿼리 추가 :

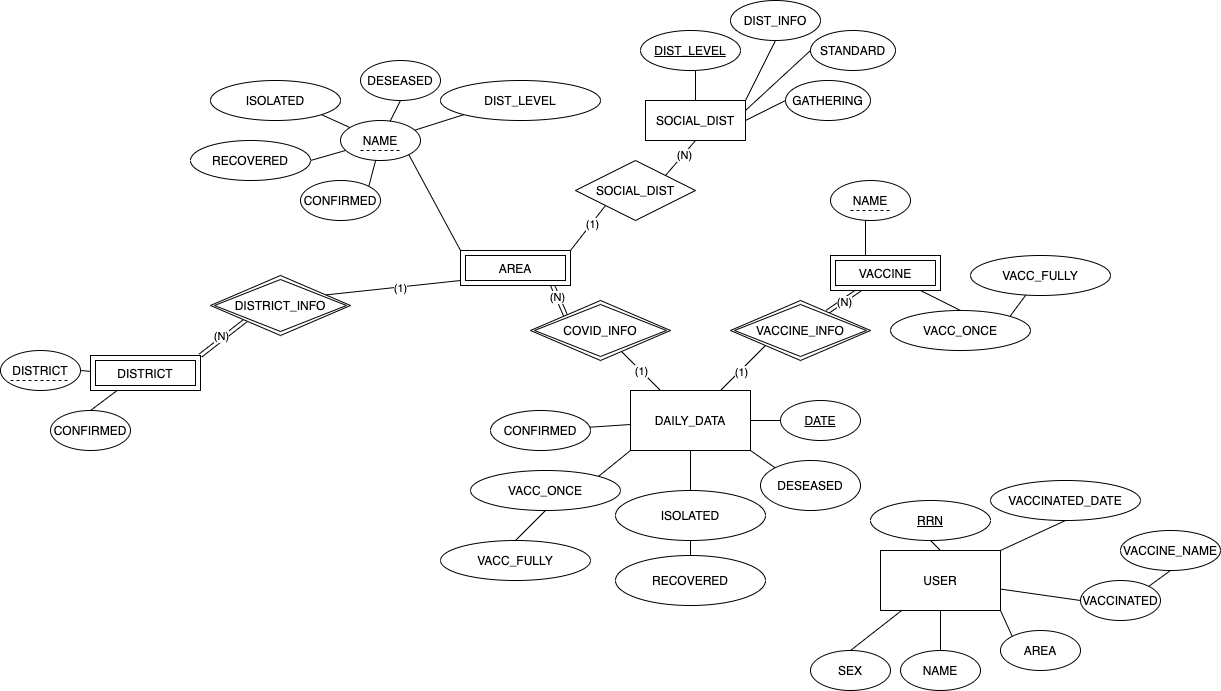
웹 어플리케이션을 통해 사용자가 속한 지역구의 전날 대비 확진자 증감 정보, 당일 확진자 정보를 제공한다. [5. WEB Application] 참조.

1. HOSPITAL 데이터 추가 :

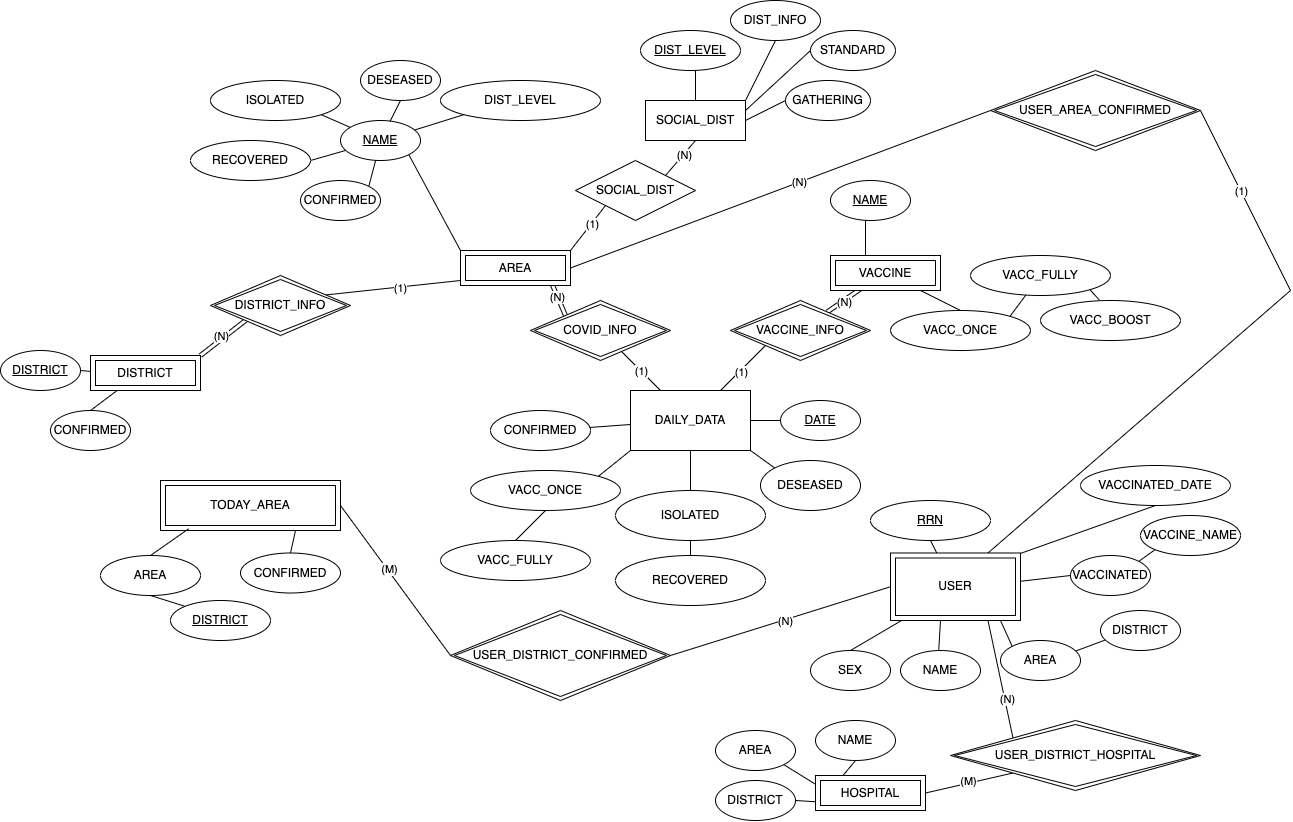
Hospital data를 크롤링해서 Database에 추가했다. 웹 어플리케이션을 통해 사용자가 속한 지역별 코로나 선별진료소를 확인할 수 있다. [5. WEB Application] 참조.



# **3.2 ER Diagram**

[수정 전]

[수정 후]



# **3.3 Data Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DAILY\_DATA Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| DATE | DATE | Y | PK | 데이터 업데이트 날짜 |
| CONFIRMED | Int | Y | - | 확진자 인원 수 |
| ISOLATED | Int | Y | - | 격리자 인원 수 |
| DESEASED | Int | Y | - | 사망자 인원 수 |
| RECOVERED | Int | Y | - | 격리 해제 인원 수 |
| VACC\_ONCE | Int | Y | - | 1차 백신 접종 인원 수 |
| VACC\_FULLY | Int | Y | - | 2차 백신 접종 인원 수 |
| VACC\_BOOST | Int | Y | - | 추가 백신 접종 인원 수 |

* 종합적인 데이터 테이블이다. primary key는 데이터 업데이트 날짜이다.
* 추가접종이 늘어나고 있어서 ‘VACC\_BOOST’를 추가했다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AREA Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| DATE | DATE | Y | PK | 데이터 업데이트 날짜 |
| NAME | Varchar | Y | PK | 지역 이름 |
| CONFIRMED | Int | Y | - | 확진자 인원 수 |
| ISOLATED | Int | Y | - | 격리자 인원 수 |
| DESEASED | Int | Y | - | 사망자 인원 수 |
| RECOVERED | Int | Y | - | 격리 해제 인원 수 |
| DIST\_LEVEL | Int | N | - | 거리두기 단계 |

* 지역별 데이터 테이블이다. primary key는 데이터 업데이트 날짜와 지역 이름이다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SOCIAL\_DIST Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| DIST\_LEVEL | Int | Y | PK | 거리두기 단계 |
| DIST\_INFO | Varchar | Y | - | 거리두기 단계 정보 |
| STANDARD | Varchar | Y | - | 거리두기 단계 기준 |
| GATHERING | Int | N | - | 거리두기 단계에 따른 모임 가능 인원 수 |

* 사회적 거리두기 데이터 테이블이다. primary key는 거리두기 단계이다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DISTRICT Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| DATE | DATE | Y | PK | 데이터 업데이트 날짜 |
| DISTRICT | Varchar | Y | PK | 자치구 이름 |
| AREA | Varchar | Y | FK | 지역 이름 |
| CONFIRMED | Int | N | - | 확진자 인원 수 |

* 자치구(구)별 데이터 테이블이다. primary key는 자치구(구) 이름, 데이터 업데이트 날짜이다. 지역 이름은 AREA 테이블의 지역 이름을 참조한 FK이다.
* 크롤링 데이터가 없는 경우, 자치구 확진 인원 수는 null값을 가질 수 있다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| VACCINE Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| DATE | DATE | Y | PK | 데이터 업데이트 날짜 |
| NAME | Varchar | Y | PK | 백신 이름 |
| VACC\_ONCE | Int | Y | - | 백신 1차 접종 인원 수 |
| VACC\_FULLY | Int | N | - | 백신 2차 접종 인원 수 |
| VACC\_BOOST | Int | N | - | 백신 추가 접종 인원 수 |

* 백신 데이터 테이블이다. primary key는 백신 이름과 데이터 업데이트 날짜이다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USER Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| NAME | Varchar | Y | - | 사용자의 이름 |
| RRN | Char(14) | Y | PK | 사용자의 주민등록번호 |
| SEX | Char(1) | Y | - | 사용자의 성별 |
| AREA | Varchar | Y | - | 사용자의 지역(시) |
| DISTRICT | Varchar | Y | - | 사용자의 지역(구) |
| VACC\_NAME | Varchar | N | - | 접종 백신 종류 |
| VACCINATED | Varchar | N | - | 사용자의 백신 접종 여부 |
| VACCINATED\_DATE | Date | N | - | 사용자의 백신 접종 날짜 |

* 사용자 데이터 테이블이다. primary key는 사용자의 주민등록 번호(RRN) 이다.
* DISTRICT 가 attribute로 추가되었다. 사용자의 지역(구)를 등록할 수 있다.
* 사용자의 성별 데이터에 따른 주민등록 식별 번호(8)가 1 또는 2로 구분되는 constraint를 갖는다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TODAY\_AREA Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| AREA | Varchar | Y | PK | 실시간 확진자 발견된 지역(시) |
| DISTRICT | Varchar | Y | PK | 실시간 확진자 발견된 지역(구) |
| CONFIRMED | Int | Y | - | 실시간 지역별 확진자 수 |

- 실시간 확진자 데이터 테이블로 새롭게 추가되었다. primary key는 오늘 확진자가 발견된 지역(시), 자치구(구)이다.

- 당일 생성되고 삭제되는 데이터이다. 유저와 연동해, 유저의 지역 - 자치구 기준 확진 정보를 알려주기 위해 활용된다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HOSPITAL Table | | | | |
| Attribute | Data Type | Not NULL | Key | 설명 |
| AREA | Varchar | Y | FK | 선별진료소 위치한 지역(시) |
| DISTRICT | Varchar | Y | PK | 선별진료소 위치한 지역(구) |
| NAME | Varchar | Y | PK | 선별진료소 이름 |

* 지역별 선별진료소 데이터 테이블이다. primary key는 선별진료소가 위치한(구), 선별진료소 이름이다.

# **4. DB Implementation**

4.1 DB Schema

|  |
| --- |
| **Schema Create 요약 코드** |
| // DAILY\_DATA TABLE  CREATE TABLE daily\_data (  update\_date date not null,  daily\_confirmed int not null,  daily\_isolated int not null,  daily\_deseased int not null,  daily\_recovered int not null,  daily\_vacc\_once int not null,  daily\_vacc\_fully int not null,  daily\_boost int not null,  primary key (update\_date),  CONSTRAINT chk\_daily\_data CHECK (daily\_confirmed >= 0  AND daily\_deseased >= 0 AND daily\_recovered >= 0  AND daily\_vacc\_once >= 0)  );  // HOSPITAL TABLE  CREATE TABLE hospital (  hospital\_area varchar(10) not null,  hospital\_district varchar(10) not null,  hospital\_name varchar(10) not null,  primary key (hospital\_name, hospital\_district),  foreign key (hospital\_area) references area (area\_name),  ); |

4.2 Constraints

* Covid19 & Vaccination information DB Data Schema 제약조건

|  |  |
| --- | --- |
| DAILY\_DATA Table | |
| Attribute | 제약조건 |
| DATE | - |
| CONFIRMED | 인원 수이므로 양수 |
| ISOLATED | 전날 인원 수를 기준으로 상대적 값이기 때문에 음수도 허용 |
| DESEASED | 인원 수이므로 양수 |
| RECOVERED | 인원 수이므로 양수 |
| VACC\_ONCE | 인원 수이므로 양수 |
| VACC\_FULLY | 인원 수이므로 양수 |
| VACC\_BOOST | 인원 수이므로 양수 |

|  |  |
| --- | --- |
| AREA Table | |
| Attribute | 제약조건 |
| DATE | - |
| NAME | 대한민국에 존재하는 행정구역 |
| CONFIRMED | 인원 수이므로 양수 |
| ISOLATED | 전날 인원 수를 기준으로 상대적 값이기 때문에 음수도 허용 |
| DESEASED | 인원 수이므로 양수 |
| RECOVERED | 인원 수이므로 양수 |
| DIST\_LEVEL | 거리두기 단계(1~4) 중의 값 |

|  |  |
| --- | --- |
| DISTRICT Table | |
| Attribute | 제약 조건 |
| DATE | - |
| DISTRICT | AREA에 존재하는 하위 행정구역 |
| AREA | 대한민국에 존재하는 행정구역 |
| CONFIRMED | 인원 수이므로 양수 |

|  |  |
| --- | --- |
| SOCIAL\_DIST Table | |
| Attribute | 제약 조건 |
| DIST\_LEVEL | 거리두기 단계(1~4) 중의 값 |
| DIST\_INFO | - |
| STANDARD | - |
| GATHERING | - |

|  |  |
| --- | --- |
| VACCINE Table | |
| Attribute | 제약 조건 |
| DATE | - |
| NAME | 백신 종류(아스트라제네카, 얀센, 화이자, 모더나) 중의 문자열 |
| VACC\_ONCE | 인원 수이므로 양수 |
| VACC\_FULLY | 인원 수이므로 양수 |
| VACC\_BOOST | 인원 수이므로 양수 |

|  |  |
| --- | --- |
| USER Table | |
| Attribute | 설명 |
| NAME | - |
| RRN | ‘-’을 포함한 13 길이의 문자열  앞 6자리는 유효한 생년월일을 나타내야 함  뒤 7자리의 첫번째 숫자는 성별을 나타내야 함 |
| SEX | 성별(M, W) 중 하나의 문자 |
| AREA | 대한민국에 존재하는 행정구역 |
| DISTRICT | 대한민국에 존재하는 하위 행정구역 |
| VACC\_NAME | 백신 종류(아스트라제네카, 얀센, 화이자, 모더나) 중의 문자열 |
| VACCINATED | 백신 접종 여부(미접종, 1차, 2차, 부스터샷 접종) 중의 문자열 |
| VACCINATED\_DATE | - |

|  |  |
| --- | --- |
| TODAY\_CONFIRMED Table | |
| Attribute | 제약 조건 |
| AREA | 대한민국에 존재하는 행정구역 |
| DISTRICT | AREA에 존재하는 하위 행정구역 |
| CONFIRMED | 인원 수이므로 양수 |

|  |  |
| --- | --- |
| HOSPITAL Table | |
| Attribute | 제약 조건 |
| AREA | 대한민국에 존재하는 행정구역 |
| DISTRICT | AREA에 존재하는 하위 행정구역 |
| NAME | DISTRICT에 존재하는 선별진료소 |

* 제약조건 구현

|  |  |
| --- | --- |
| **제약 조건** | **내용 및 주요 코드** |
| 인원 | 인원과 관련한 데이터는 0과 같거나 그보다 커야 한다. |
| 격리자의 경우 인원이 줄어들 수 있어 음의 정수의 값도 허용한다. |
| //DAILY\_DATA  CONSTRAINT chk\_daily\_data CHECK (daily\_confirmed >= 0  AND daily\_deased >= 0 AND daily\_recovered >= 0 AND daily\_vacc\_once >= 0)  //AREA  CONSTRAINT chk\_area CHECK (area\_confirmed >= 0  AND area\_deseased >= 0 AND area\_recovere d >= 0 AND area\_dist\_level > 0)    //DISTRICT  CONSTRAINT chk\_district CHECK (district\_confirmed >= 0)  //TODAY\_AREA  CONSTRAINT chk\_today\_confirmed CHECK (today\_confirmed >= 0)  //VACCINE  CONSTRAINT chk\_vaccine CHECK (vacc\_once >= 0) | |
| 지역명 | area의 경우 대한민국에 존재하는 행정구역명이여야 한다. |
| district의 경우 area에 존재하는 하위 행정구역명이여야 한다. |
| //AREA  CONSTRAINT chk\_area\_name CHECK (area\_name = '서울' OR area\_name = '부산' OR area\_name = '대구' OR area\_name = '인천' OR area\_name = '광주' OR area\_name = '대전' OR area\_name = '울산' OR area\_name = '세종' OR area\_name = '경기' OR area\_name = '강원' OR area\_name = '충북' OR area\_name = '충남' OR area\_name = '전북' OR area\_name = '전남' OR area\_name = '경북' OR area\_name = '경남' OR area\_name = '제주')  );  //DISTRICT  CONSTRAINT chk\_district\_name CHECK (  (area\_name = '서울' AND (district = '성북구' OR district = '은평구' OR district = '송파구' OR district = '노원구' OR district = '관악구' OR district = '강동구' OR district = '양천구' OR district = '도봉구' OR district = '동작구' OR district = '광진구' OR district = '강남구' OR district = '중랑구' OR district = '서대문구' OR district = '마포구' OR district = '구로구' OR district = '용산구' OR district = '성동구' OR district = '종로구' OR district = '중구' OR district = '동대문구' OR district = '강북구' OR district = '강서구' OR district = '금천구' OR district = '영등포구' OR district = '서초구'))  OR (area\_name = '부산' AND (district = '중구' OR district =  ```생략```  ); | |
| 거리두기 | 거리두기 단계는 반드시 양의 정수의 단계를 가져야 한다. |
| CONSTRAINT chk\_social\_dist CHECK (dist\_level > 0) | |
| 백신 | 백신의 이름은 현재 접종이 가능한 백신이여야한다.  (아스트라제네카, 얀센, 화이자, 모더나) |
| 백신 접종 여부는 미접종, 1차접종, 2차접종, 부스터샷 접종 이 4가지로 이루어진다. |
| CONSTRAINT chk\_vaccine\_name  CHECK (vacc\_name = '화이자' OR vacc\_name = '모더나' OR vacc\_name = '아스트라제네카' OR vacc\_name = '얀센') | |
| 사용자 | 사용자의 성별과 주민등록번호의 성별 확인 데이터가 일치해야한다 |
| CONSTRAINT chk\_user CHECK ((user\_sex = 'M' AND substr(user\_rrn, 8,1) = 1)  OR (user\_sex = 'M' AND substr(user\_rrn, 8,1) = 3)  OR (user\_sex = 'W' AND substr(user\_rnn, 8, 1) = 2)  OR (user\_sex = 'W' AND substr(user\_rnn, 8, 1) = 4)) | |

4.3 Case scenarios for Requirements

|  |
| --- |
| **DATA INSERT 주요 코드** |
| // HOSPITAL (크롤링 데이터 파일 대량 인서트)  const fs = require("fs");  const Hospital\_jsonFile = fs.readFileSync('../크롤링 데이터/Hospital.json', 'utf8');  const Hospital\_jsonData =JSON.parse(Hospital\_jsonFile.toString().trim());  const hospital\_result = Hospital\_jsonData.HOSPITAL.map(v => `INSERT INTO hospital(hospital\_area, hospital\_district, hospital\_name)  VALUES('${v.AREA}','${v.DISTRICT}','${v.NAME}');`).join('\n');  fs.writeFileSync("hospital.txt", hospital\_result, { encoding: "utf8" }); |
| // DAILY\_DATA TABLE (위의 값 SQL QUERY 샘플)  INSERT INTO daily\_data(update\_date, daily\_confirmed, daily\_isolated, daily\_deseased, daily\_recovered, daily\_vacc\_once, daily\_vacc\_fully, daily\_boost)  VALUES('2021-02-26','6010',-788,583,1183,28175,5008,0);    // DISTRICT TABLE  INSERT INTO district(update\_date, area\_name, district, district\_confirmed)  VALUES('2021-11-30','제주','제주',31); |
| 텍스트이(가) 표시된 사진  자동 생성된 설명텍스트이(가) 표시된 사진  자동 생성된 설명  텍스트이(가) 표시된 사진  자동 생성된 설명 텍스트이(가) 표시된 사진  자동 생성된 설명  텍스트이(가) 표시된 사진  자동 생성된 설명 텍스트이(가) 표시된 사진  자동 생성된 설명  텍스트이(가) 표시된 사진  자동 생성된 설명 |

|  |  |
| --- | --- |
| **SQL QUERY[[1]](#footnote-1)** | |
| Insert | //사용자 회원가입  INSERT INTO user SET **(이름,주민번호,지역,자치구,성별,백신종류,접종여부,접종날짜)** |
| Delete | //지역별 선별진료소 삭제(hospital)  //실시간 지역 확진자수 하루 기준으로 초기화(today\_confirmed)  //회원 탈퇴(user)  DELETE FROM user WHERE user\_rrn = **유저주민등록번호** |
| Update | //지역 거리두기 단계 변경(area)  //회원 백신 접종 정보 변경(user)  //실시간 지역 확진자수 변경(today\_confirmed) |
| Retrieval | // 유저 근처 선별진료소 확인 (JOIN 사용)  SELECT hospital\_name FROM user INNER JOIN hospital ON user\_district = hospital\_district AND hospital\_area = user\_area  // 유저 지역의 확진자 수 (GROUP BY, JOIN 사용)  SELECT sum(today\_confirmed)  FROM user INNER JOIN today\_area ON user\_area = today\_area  WHERE user\_rrn = **주민번호** GROUP BY today\_area  // 날짜별 데이터 확인  // 🡪 두개 날짜 사이, 지역의 코로나 데이터의 평균 값을 가져온다. (GROUP BY , HAVING)  SELECT area\_name, avg(SELECT area\_name, avg(area\_confirmed), avg(area\_isolated), avg(area\_deseased), avg(area\_recovered), ROUND(avg(area\_dist\_level))  FROM area WHERE update\_date BETWEEN **날짜1** AND **날짜2**  GROUP BY area\_name HAVING area\_name = **지역**  //(ODER BY 사용) |

4.4 QUERY PROCESSING OPTIMIZATION

* **GROUP BY**
* 데이터 특성상 종류별로 묶을 경우가 많이 생겨 GROUP BY를 많이 사용하였고 추가적인 조건이 필요할 경우 HAVING을 이용하여 원하는 데이터를 가져왔다.

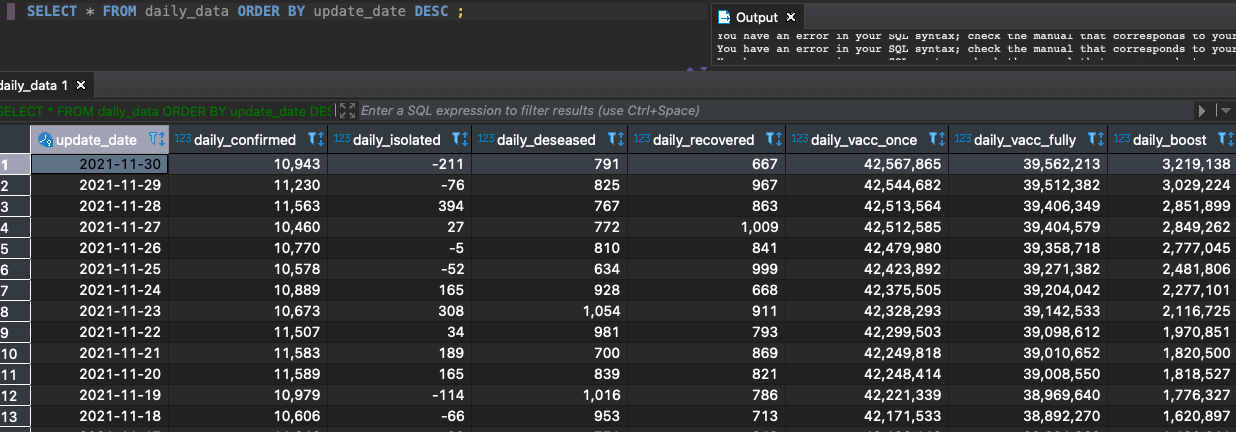
.

* GROUP BY와 HAVING을 동시에 사용하게 되면 GROUP BY로 모든 데이터들을 다 가져온 뒤 HAVING으로 불필요한 데이터를 처리하기 때문에 리소스 측면에서 비효율적이라 최대한 WHERE을 이용했다.
* **INNER JOIN**
* 실시간 데이터와 사용자의 데이터와 연관짓기 위해 INNER JOIN을 이용했다.
* INNER JOIN이 아닌 JOIN의 경우 해당 테이블을 기준으로 JOIN을 진행해 원하는 데이터를 얻어낼 수 없다.
* INNER JOIN이 여러 JOIN들 중 속도적인 측면에 강점을 가진다.

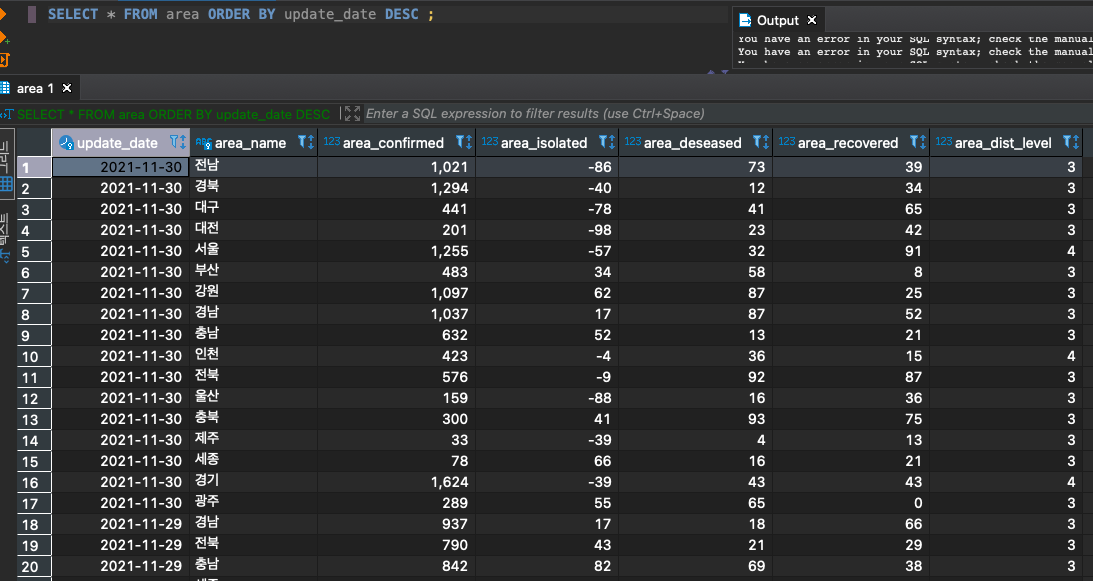
5. Result

5.1 Relational model

① DAILY\_DATA



② AREA



③ SOCIAL\_DIST

텍스트, 스크린샷, 모니터, 화면이(가) 표시된 사진

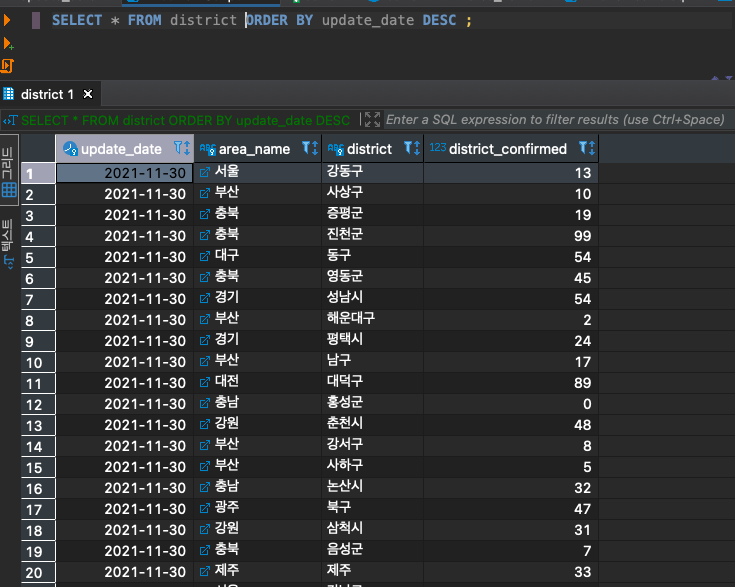
자동 생성된 설명

⑤ VACCINE

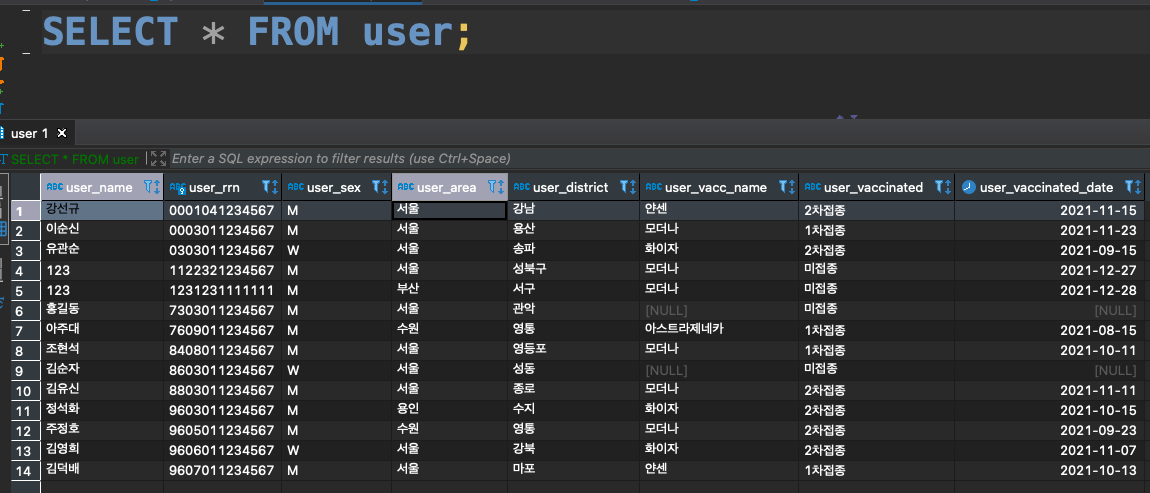
텍스트이(가) 표시된 사진

자동 생성된 설명

⑥ DISTRICT



⑦ USER

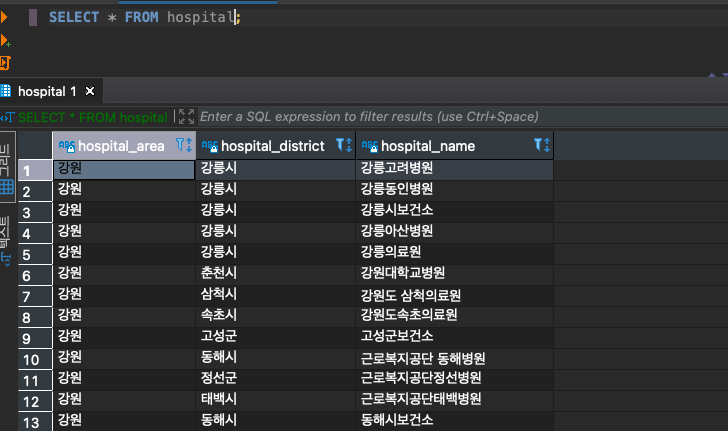


⑧ TODAY\_AREA

텍스트, 전자기기, 컴퓨터, 스크린샷이(가) 표시된 사진

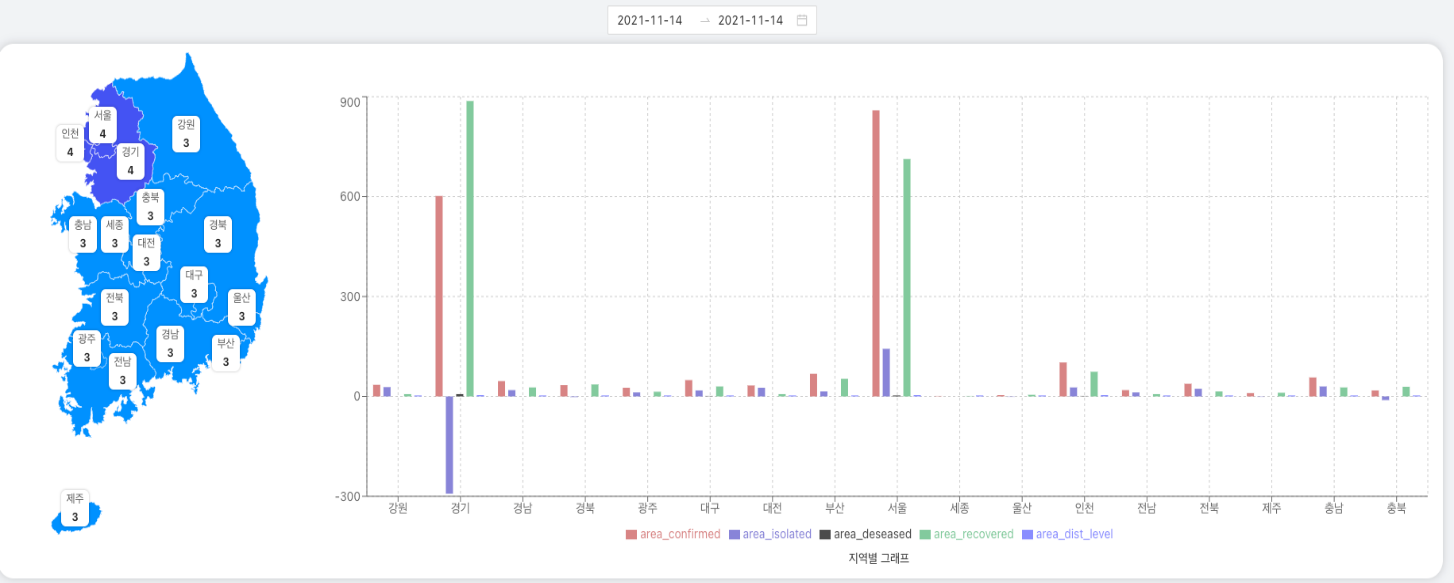
자동 생성된 설명

⑨ HOSPITAL

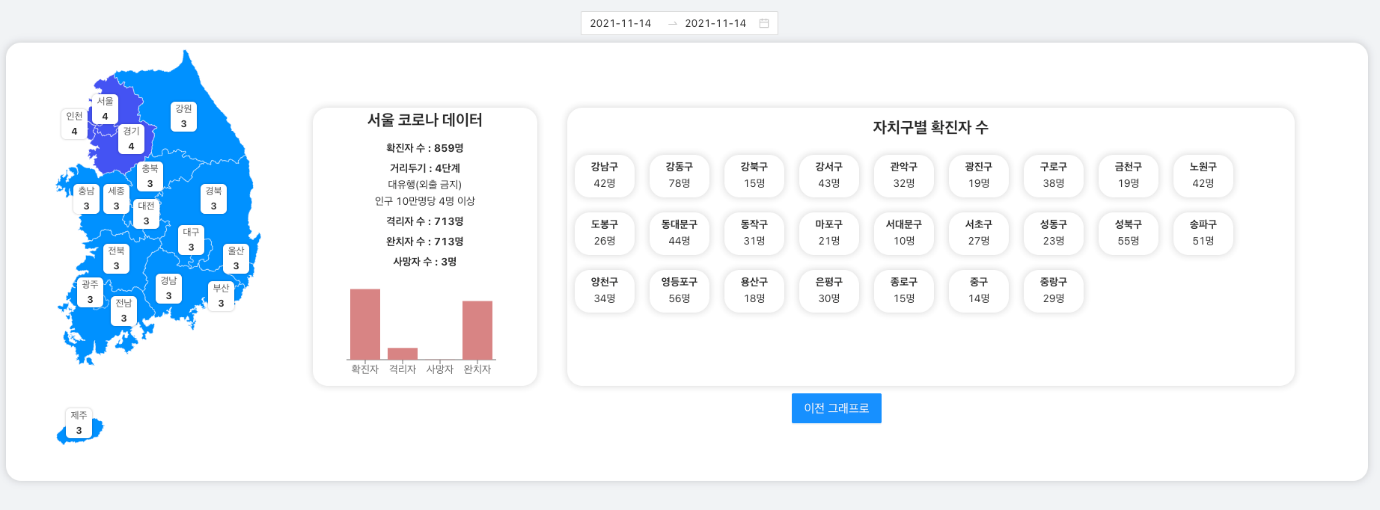


5.2 WEB Application

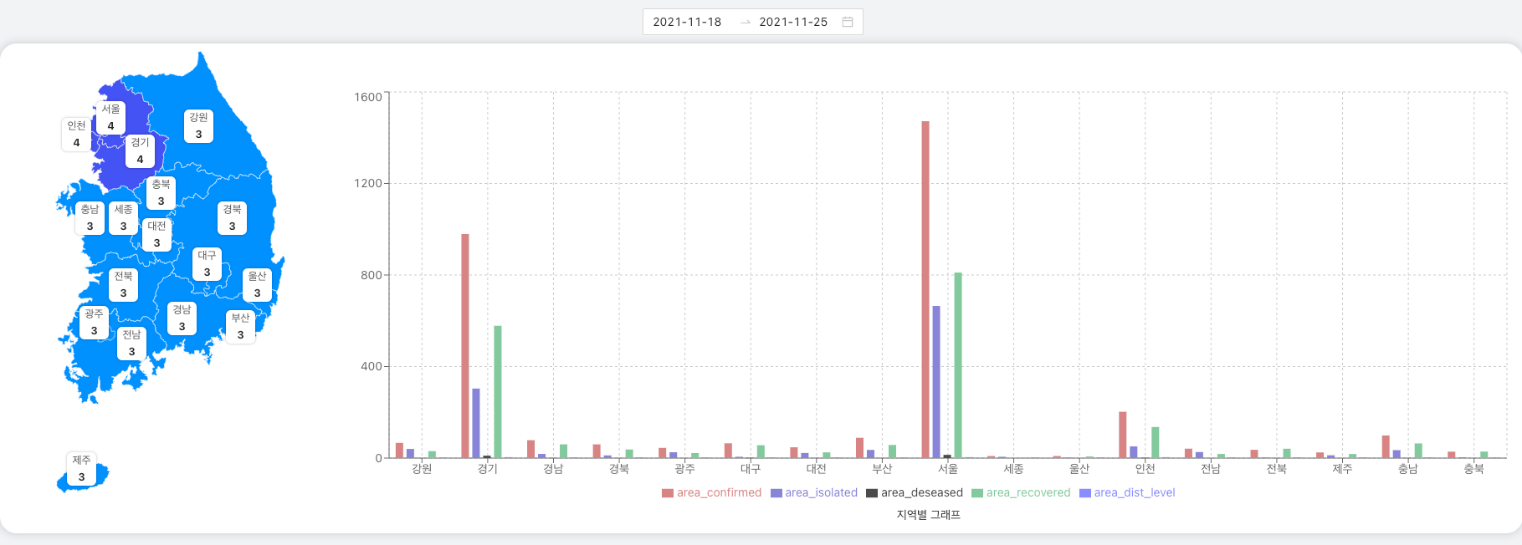
|  |  |
| --- | --- |
| Case 1 | 지역 별 데이터 🡪 날짜 하나 선택 |
| 해당 날짜의 모든 지역의 코로나 데이터를 가져온다.  SELECT \* FROM area WHERE update\_date = **날짜** | |



|  |  |
| --- | --- |
| Case 2 | 지역 별 데이터 🡪 날짜 하나 선택 후 지역 선택 |
| 1. 해당 지역 코로나 데이터를 가져온다.   SELECT \* FROM area WHERE update\_date = **날짜** | |
| 1. 해당 지역의 자치구 확진자 데이터를 가져온다.   SELECT \* FROM district WHERE update\_date = **날짜** AND area\_name = **지역** | |



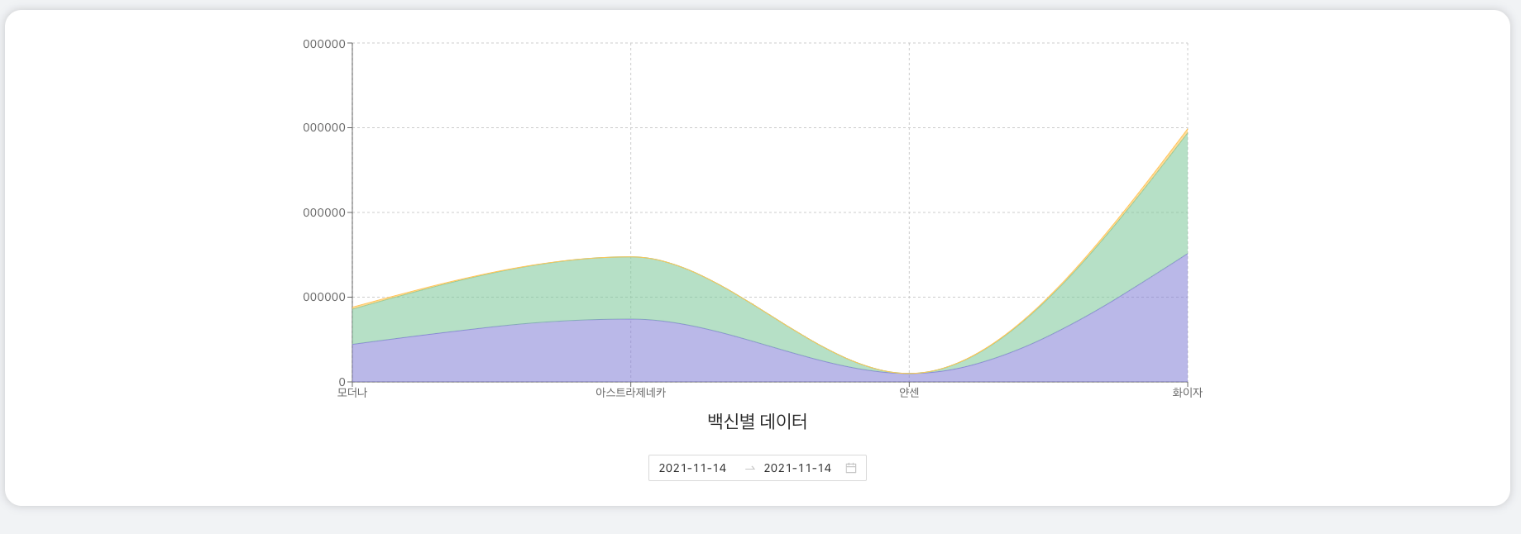
|  |  |
| --- | --- |
| Case 3 | 지역 별 데이터 🡪 날짜 두개 선택 |
| 두개 날짜 사이의 전국 코로나 데이터의 평균 데이터를 가져온다. (GROUP BY 사용)  SELECT area\_name, avg(area\_confirmed), avg(area\_isolated), avg(area\_deseased), avg(area\_recovered), ROUND(avg(area\_dist\_level))  FROM area  WHERE update\_date BETWEEN **날짜1** AND **날짜2** GROUP BY area\_name | |



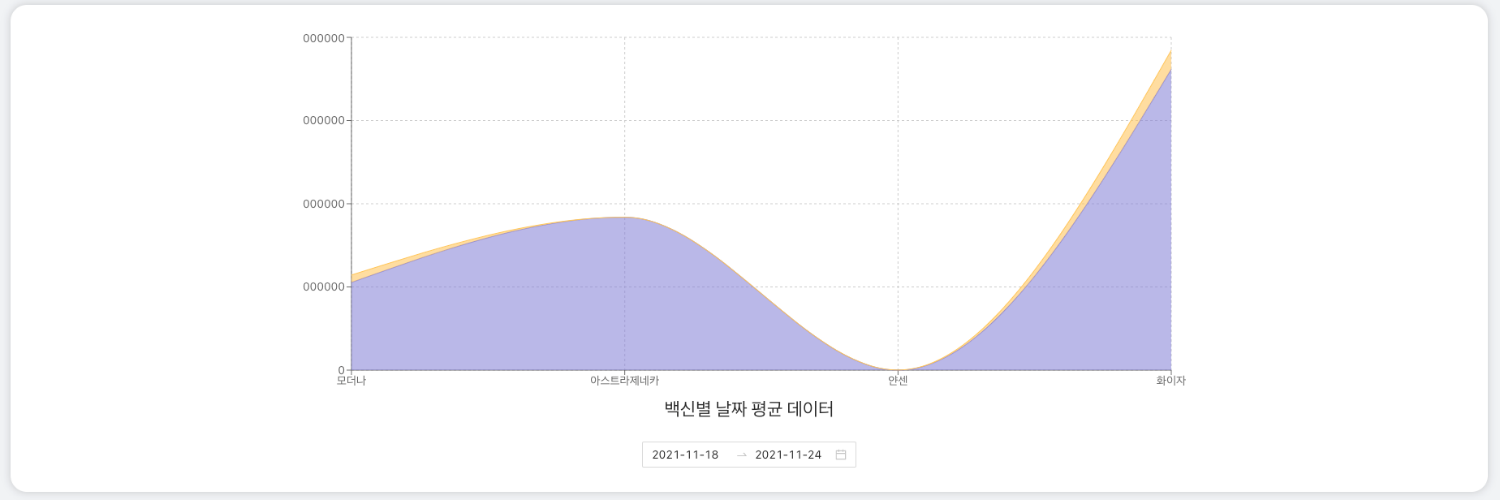
|  |  |
| --- | --- |
| Case 4 | 지역 별 데이터 🡪 날짜 두개 선택 후 지역 선택 |
| 1. 두개 날짜 사이, 지역의 코로나 데이터의 평균 값을 가져온다. (GROUP BY 사용)   SELECT area\_name, avg(SELECT area\_name, avg(area\_confirmed), avg(area\_isolated), avg(area\_deseased), avg(area\_recovered), ROUND(avg(area\_dist\_level))  FROM area WHERE update\_date BETWEEN **날짜1** AND **날짜2**  GROUP BY area\_name HAVING area\_name = **지역** | |
| 1. 두개 날짜 사이, 자치구의 코로나 데이터의 평균 값을 가져온다. (GROUP BY 사용)   SELECT district , avg(district\_confirmed) as district\_confirmed FROM district WHERE update\_date BETWEEN **날짜1** AND **날짜2** AND area\_name = **지역** GROUP BY district | |

이미지

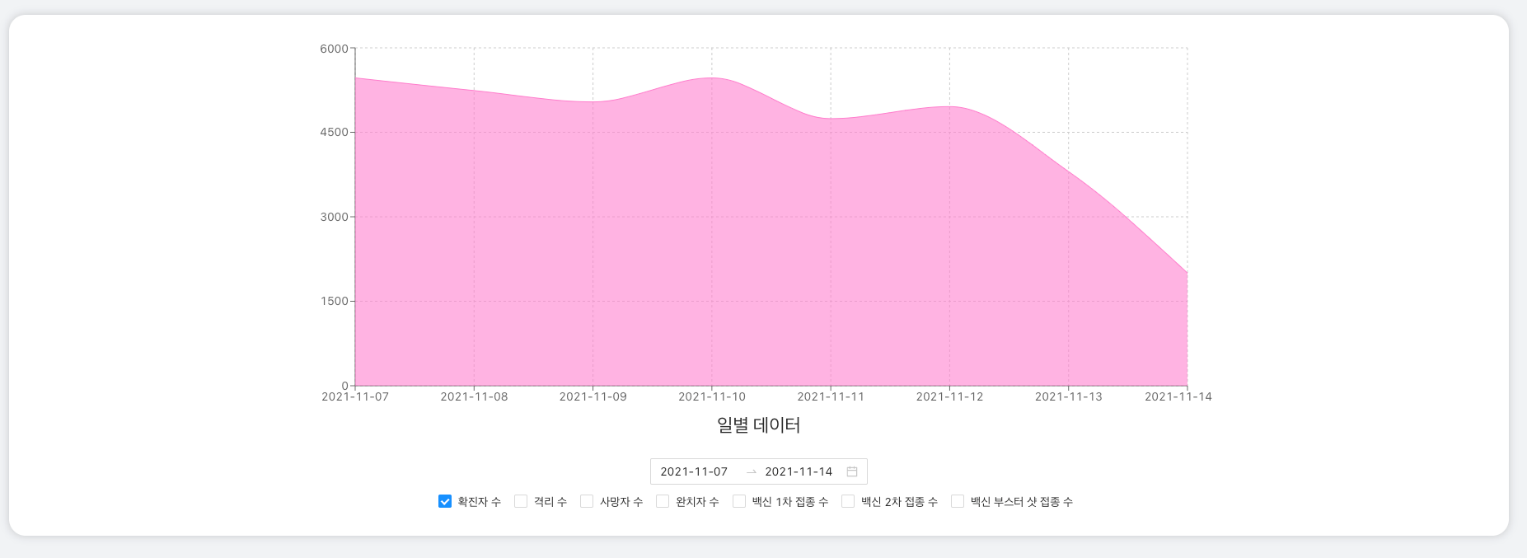
|  |  |
| --- | --- |
| Case 5 | 백신 별 데이터 🡪 날짜 하나 선택 |
| 해당 날짜의 백신 데이터를 가져온다.  SELECT \* FROM vaccine WHERE update\_date = **날짜** | |



|  |  |
| --- | --- |
| Case 6 | 백신별 데이터 🡪 날짜 두개 선택 |
| 날짜 사이 데이터의 평균 데이터를 가져온다. (GROUP BY 사용)  SELECT vacc\_name, avg(vacc\_once), avg(vacc\_fully), avg(vacc\_boost)  FROM vaccine WHERE update\_date BETWEEN **날짜1** AND **날짜2** GROUP BY vacc\_name | |

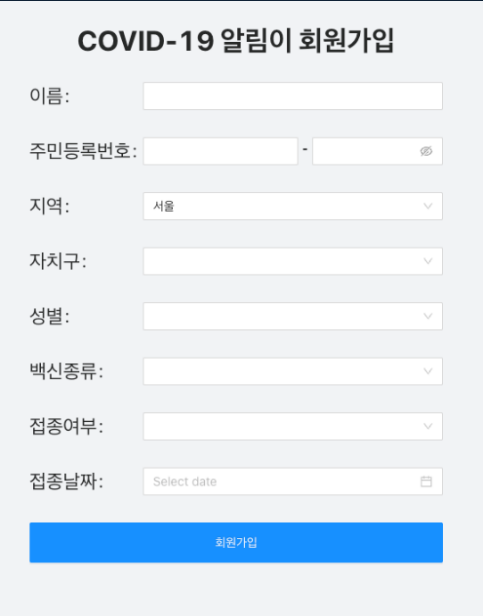
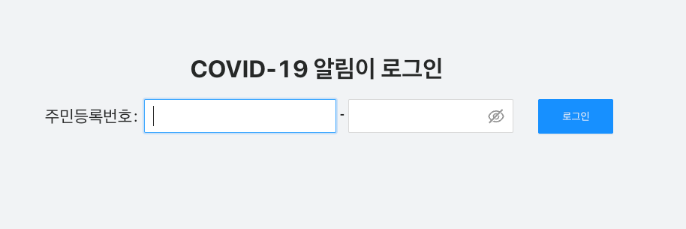


|  |  |
| --- | --- |
| Case 7 | 일별 데이터 🡪 날짜 선택 |
| 1. 날짜 하나 선택해서 일별 총 데이터를 가져온다.   SELECT **(체크박스에서 선택한 수)** FROM daily\_data WHERE update\_date = **날짜** | |
| 1. 날짜 두개 선택해서 일별 총 데이터를 가져온다.   SELECT **(체크박스에서 선택한 열)** FROM daily\_data WHERE update\_date  BETWEEN **날짜1** AND **날짜2** | |

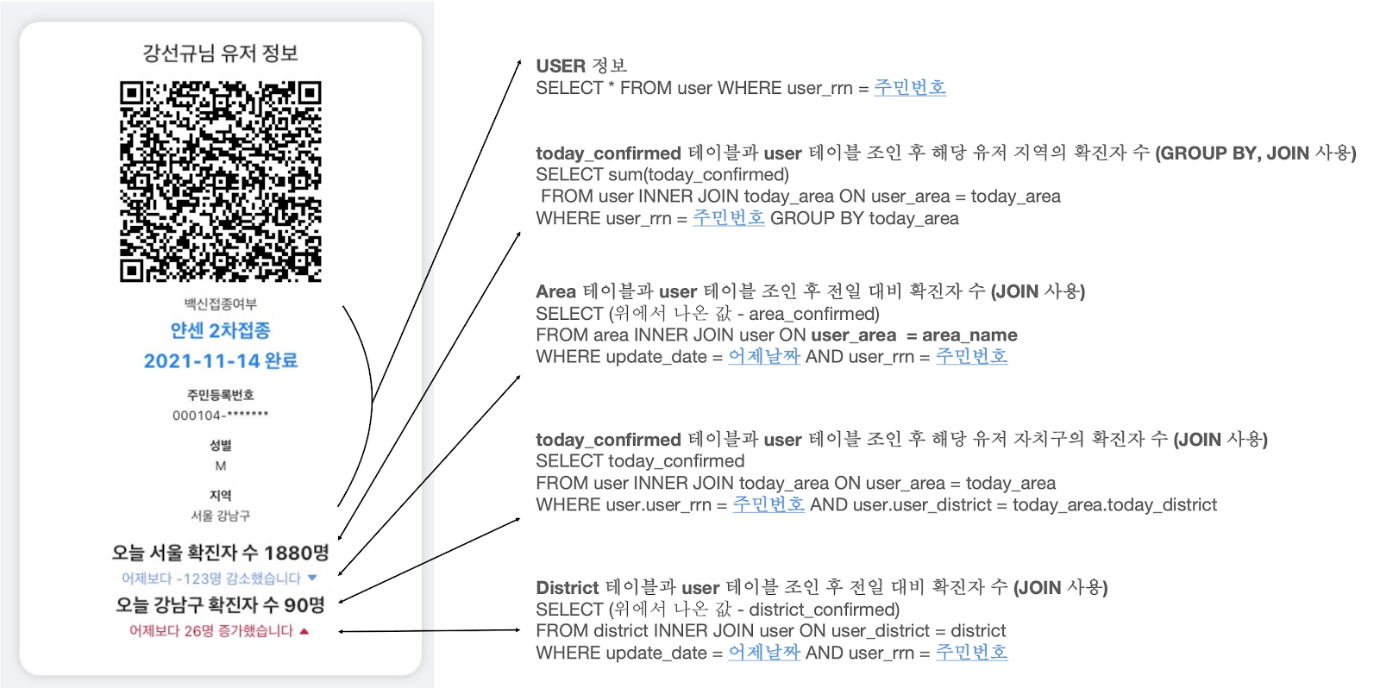


|  |  |
| --- | --- |
| Case 8 | 로그인 |
| 해당 데이터 존재한다면 로그인 처리  SELECT \* FROM user WHERE user\_rrn = **유저주민번호** | |

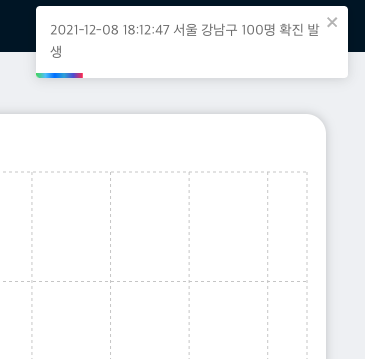
|  |  |
| --- | --- |
| Case 9 | 회원가입 |
| INSERT INTO user SET **(이름,주민번호,지역,자치구,성별,백신종류,접종여부,접종날짜)** | |



|  |  |
| --- | --- |
| Case 10 | USER🡪 유저 지역구 데이터 확인 |



|  |  |
| --- | --- |
| Case 11 | 유저 로그인 시 알림 |
| SELECT today\_confirmed FROM user INNER JOIN today\_area  ON user\_area = today\_area AND user\_district = today\_district  WHERE user.user\_rrn = **유저주민번호** | |



# **6. Crawling Data & DevOps**

# **6.1 Crawling Data**

|  |  |
| --- | --- |
| Table | 크롤링한 사이트 URL |
| DAILY\_DATA | <https://corona-live.com/>  https://ncv.kdca.go.kr/vaccineStatus.es?mid=a11710000000 |
| AREA | <https://kosis.kr/covid/covid_index.do>  http://ncov.mohw.go.kr/regSocdisBoardView.do |
| DISTRICT | https://www.seoul.go.kr/coronaV/coronaStatus.do |
| VACCINE | https://ncv.kdca.go.kr/vaccineStatus.es?mid=a11710000000 |
| DIST\_LEVEL | http://ncov.mohw.go.kr/socdisBoardList.do?brdId=6&brdGubun=64&dataGubun=641 |
| TODAY\_CONFIRMED | https://corona-live.com/ |
| HOSPITAL | https://www.mohw.go.kr/react/popup\_200128\_3.html |

**DAILY\_DATA**



당일 날의 코로나 확진자 수, 격리자 수, 완치자 수, 사망자 수를 크롤링하였다.

텍스트이(가) 표시된 사진

자동 생성된 설명

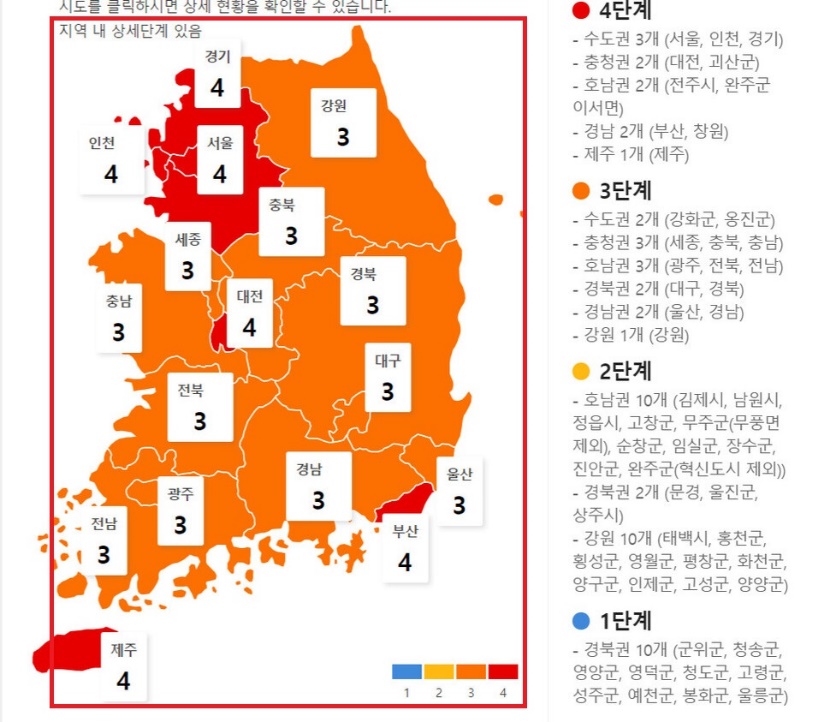
당일 날의 코로나 백신 1차 접종 수, 2차 접종 수, 부스터 샷 접종 수를 크롤링하였다.

**AREA**

지도이(가) 표시된 사진

자동 생성된 설명

당일 날 지역별 확진자 수, 완치자 수, 사망자 수, 격리자 수를 크롤링하였다.



당일 날 지역별 거리두기 단계를 크롤링하였다.

**DISTRICT**

텍스트이(가) 표시된 사진

자동 생성된 설명

당일 날 자치구별 확진자 수를 크롤링하였다.

**VACCINE**



당일 날 백신별 차수별 접종자 수를 크롤링하였다.

**DIST\_LEVEL**

테이블이(가) 표시된 사진

자동 생성된 설명

거리두기 단계별 정보를 크롤링하였다.

**TODAY\_CONFIRMED**

텍스트이(가) 표시된 사진

자동 생성된 설명

당일 날 지역 별로 발생한 확진자 수를 실시간으로 크롤링하였다.

**HOSPITAL**

텍스트이(가) 표시된 사진

자동 생성된 설명

자치구 별로 위치하는 코로나 선별진료소를 크롤링하였다.

# **6.2 DevOps**

|  |  |
| --- | --- |
| **사용 언어** | JavaScript |
| **Frontend개발 도구** | React |
| **Backend 개발도구** | Node JS, Express |
| **데이터베이스** | MySQL |
| **버전 관리** | Git, GitHub |

텍스트이(가) 표시된 사진

자동 생성된 설명

해당 프로젝트는 위 스택을 기반으로 웹 어플리케이션으로 개발되었다.

결과물은 **5.** 내용에서 확인할 수 있다.

Node.js와 MySQL을 연결하여 물리적 설계를 진행하고 연결된 MySQL에 Express 프레임워크 REST API와 연결해 DML을 요청 후 데이터를 조회/수정할 수 있다.

클라이언트는 전 단계에서 구현된 REST API를 통해 데이터 조회를 할 수 있다.

# 7. 참고자료

1. [https://github.com/jujeongho0/db.git 참조.] [↑](#footnote-ref-1)