

REPORT



데이터베이스및데이터시각화

< Project #2 – Project 제안서 >

제출일자 : 2020년 10월 29일

담당교수 : 이기훈교수님

학과 : 컴퓨터정보공학과

이름 : 강성운 - 2017202090

류화랑 - 2017202082

임경택 - 2017202036

조정우 - 2017202084

■ Project Overview

코로나 백신 예약 사이트를 직접 구현해본다.

이때 단순히 모양만 따라하는 것이 아니라, ER 다이어그램을 이용해 체계적으로 설계 후 구현 시 백 엔드 단계에는 SQL을 활용한 DBMS가 동작하는 실사용 가능한 데이터베이스와 이를 기반으로 한 시각적인 웹 화면을 구성하는 것이 궁극적인 목표이다.

1단계 | 예약정보 입력

2단계 | 예약완료 안내

개인정보 수집 및 이용에 대한 안내





「감염병의 예방 및 관리에 관한 법률」 제33조의 4 및 같은 법 시행령 제32조의 3에 따라
코로나19 예방접종 예약신청을 위하여 다음의 개인정보를 수집합니다.

※ 본인 인증된 사용자를 변경하거나, 이전화면으로 돌아가면 인증 및 대기열을 다시 진행하셔야 합니다.

본인/대리 예약 구분 *	<input checked="" type="radio"/> 본인예약 <input type="radio"/> 대리예약
접종받는 분(피접종자) 이름 *	<input type="text"/>
접종받는 분(피접종자) 주민등록번호 *	<input type="text"/> - <input type="text"/> <button>대상자 확인</button>
예약대상 백신	-
예약대상자 구분	-
휴대폰번호 *	<input type="text"/> - <input type="text"/> - <input type="text"/> ※ 예약정보는 휴대폰 번호로 전송해드리오니, 확인 후 예약일시에 맞춰 의료기관에 방문해주세요.
국민비서 알림 선택 *	백신접종 일시, 장소, 주의사항 등을 휴대폰으로 알려드립니다. <input type="radio"/> 네이버앱 <input type="radio"/> 카톡(카카오톡) <input type="radio"/> 토스(toss) <input type="radio"/> 문자메시지 <input type="radio"/> 수신받지 않음
의료기관 및 예약일시 선택 *	<button>의료기관 찾기</button> <button>예약</button> <button>취소</button>

※ 부득이한 사정으로 인하여 예약일시에 백신접종을 진행하지 못한 경우, 접종 미완료가 확인될 때까지 예약이 불가능하오니 신중하게 선택해주세요.

■ Development environment

DBMS	Framework	Language	Library
			

■ Function description

A. 홈페이지 기본

- 회원가입 및 개인 정보 조회
- 일반 이용객(백신 접종 예정자), 의사, 정부 접속 페이지 분리

B. 백신 예약

- 백신 예약 기능
 - = 접종 날짜 및 시간, 병원 선택
- 백신 종류 확인, 예약 시간, 장소 확인
- 병원 별 잔여 백신 현황
 - = 잔여 백신의 경우, 효력이 남아있을 수 있는 개봉 후 12시간 이내인지 체크
- 접종결과
- 백신 1차 접종 시 2차 자동 예약
 - = 1차접종 후 일괄로 4주 후 2차 접종으로 예약
 - = 접종 후 14일 이내 예약 제한

C. 통계

- 날짜 별 접종 추이 그래프
- 연령대 별 접종 파이 그래프

■ Web design

병원 검색

투여하는 백신 종류와 지역을 선택하면 아래에 병원 목록이 나오고 병원 선택 시 해당 병원의 정보와 진료시간을 보여준다.

The image displays two overlapping wireframe mockups of a web application for hospital search.

Top Mockup (Search Interface):

- Navigation bar: 홈페이지 로고, 백신 예약, 예약 조회/변경, 병원 검색, 게시판, 로그인
- Section header: 병원검색
- Search filters: 백신 명, 시/도, 군/구, 읍/면/동, 검색
- Search results list (left): @@의원, ##내과, **보건소, ..., === 가정의원

Bottom Mockup (Results Page):

- Navigation bar: 홈페이지 로고, 백신 예약, 예약 조회/변경, 병원 검색, 게시판, 로그인
- Section header: 병원검색
- Search filters: 백신 명, 시/도, 군/구, 읍/면/동, 검색
- Search results list (left): @@의원, ##내과, **보건소, ..., === 가정의원
- Selected hospital details (right):
 - ## 내과 (잔여백신 예약)
 - 병원 정보: 주소 : 00시 00구 00로 000 (\$\$빌딩, 107호), 전화 번호 : 02-000-0000
 - 진료시간: 월요일 : 10:00 ~ 17:00, 화요일 : 10:00 ~ 17:00, 수요일 : 10:00 ~ 17:00, 목요일 : 10:00 ~ 17:00, 금요일 : 10:00 ~ 17:00, 토요일 : 10:00 ~ 14:00, 일요일 : 휴진

로그인/회원 가입

The image displays two wireframe mockups for the login and registration pages.

로그인 (Login):

- Section header: 로그인
- Form fields: 이메일, 비밀번호
- Button: 로그인
- Link: [계정이 없으신가요?](#)

회원가입 (Registration):

- Section header: 회원가입
- Form fields: 이메일, 비밀번호, 휴대폰번호, 성함, 주민번호
- Button: 회원가입
- Link: [이메일 계정이 있으신가요?](#)

백신 예약

인적 사항을 기재한 후 병원과 일시를 선택하고 예약한다. 병원 선택시에는 modal 창을 추가로 화면에 생성해서 병원에 대한 정보와 예약 가능일시를 선택한다.

홈페이지 로고
백신 예약
예약 조회/변경
병원 검색
계시판
로그인

백신예약

이름	<input type="text"/>
주민번호	<input type="text"/> - <input type="text"/>
전화 번호	<input type="text"/> - <input type="text"/> - <input type="text"/>
백신 종류	-
알림 방식 선택	<input checked="" type="radio"/> SMS <input type="radio"/> E-MAIL <input type="radio"/> 카카오톡 <input type="radio"/> 수신 받지 않음
의료기관 및 예약일시 선택	<div>병원선택</div>

예약
취소

홈페이지 로고
백신 예약
예약 조회/변경
병원 검색
계시판
로그인

백신예약

병원 예약 가능 일자 조회

날짜 선택 (YYYY-MM-DD)
시/도
군/구
읍/면/동
기관명
검색

홈페이지 로고
백신 예약
예약 조회/변경
병원 검색
계시판
로그인

백신예약

병원 예약 가능 일자 조회

2021.10.26
서울특별시
노원구
월계동
기관명
검색

@@의원
##내과
**보건소
&& 소아 청소년 과의원
^^^이비인후과 의원
=== 가정의원

홈페이지 로고
백신 예약
예약 조회/변경
병원 검색
계시판
로그인

백신예약

병원 예약 가능 일자 조회

2021.10.26
서울특별시
노원구
월계동
기관명
검색

@@의원
##내과
**보건소
&& 소아 청소년 과의원
^^^이비인후과 의원
=== 가정의원

병원 정보
주소 : 00시 00구 00로 000
(\$\$빌딩, 107호)
전화 번호 : 02-000-0000

진료시간
월요일 : 10:00 ~ 18:00
- 점심시간 : 12:00 ~ 13:00
화요일 : 10:00 ~ 18:00
- 점심시간 : 12:00 ~ 13:00
수요일 : 10:00 ~ 18:00
- 점심시간 : 12:00 ~ 13:00
목요일 : 10:00 ~ 18:00
- 점심시간 : 12:00 ~ 13:00
금요일 : 10:00 ~ 18:00
- 점심시간 : 12:00 ~ 13:00
토요일 : 10:00 ~ 14:00
일요일 : 휴진

10:00 11:00 13:00 14:00
15:00 16:00 17:00 18:00

예약 조회

예약조회 페이지에서는 이름, 주민번호, 예약 번호를 넣으면 정보 조회 및 예약 변경/취소가 되도록 구성한다. 변경을 누르면 백신 예약 페이지로 넘어가 다시 예약을 할 수 있게 한다. 취소를 누를 경우 경고창을 생성해 한번 더 확인한 다음에 예약을 취소한다.

홈페이지 로고

백신 예약

예약 조회/변경

병원 검색

게시판

로그인

예약조회

이름	<input type="text"/>
주민번호	<input type="text"/> - <input type="text"/>
예약 번호	<input type="text"/>

예약취소

홈페이지 로고

백신 예약

예약 조회/변경

병원 검색

게시판

로그인

예약조회

예약자 정보

이름	○○○
주민번호	970000-1000000
전화 번호	010-○○○○-○○○○

예약자 정보

예약 번호	12341234
1차 접종 의료 기관	##내과
1차 접종일시	2021-10-26 / 15:00
2차 접종 의료 기관	##내과
2차 접종일시	2021-11-23 / 15:00

예약 변경예약 취소확인

홈페이지 로고

백신 예약

예약 조회/변경

병원 검색

게시판

예약조회

예약자 정보

이름	○○○
주민번호	970000-1000000
전화 번호	

예약자 정보

예약 번호	
1차 접종 의료 기관	
1차 접종일시	2021-10-26 / 15:00
2차 접종 의료 기관	##내과
2차 접종일시	2021-11-23 / 15:00

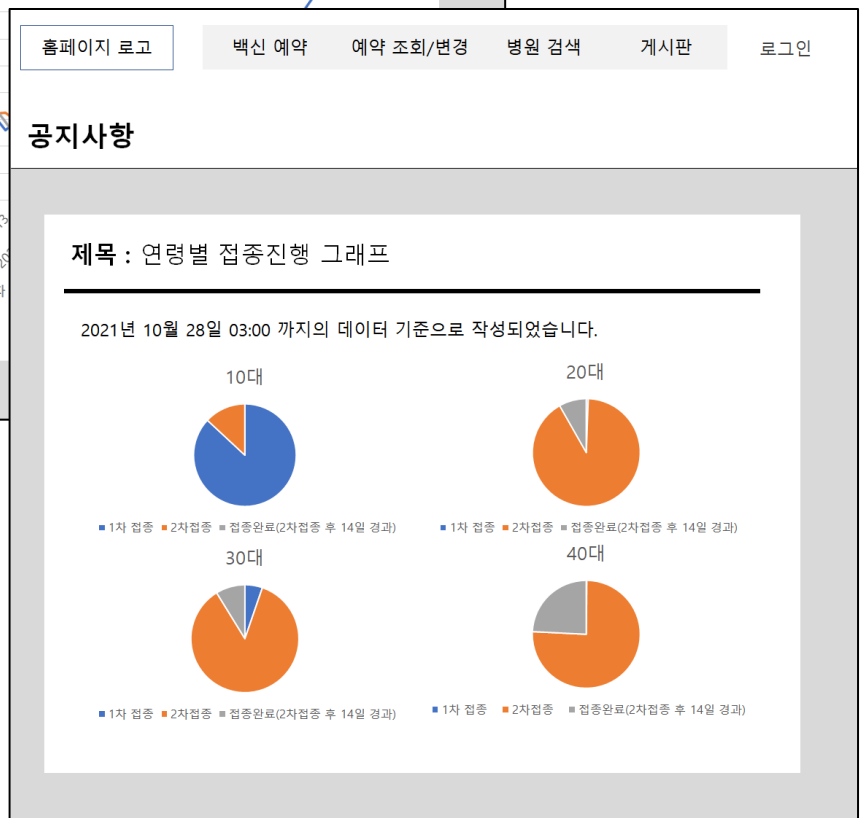
예약 변경예약 취소확인

정말로 예약을 취소하겠습니까?

예아니요

통계 그래프

SQL로 정리된 데이터를 React를 통해 그래프로 표현 해서 출력한다.



병원 관리 페이지

담당의사가 환자 예약 상태 조회 및 수정을 한다.

[홈페이지 로고](#)[예약검색](#)[접종완료자 현황](#)[공지사항](#)[게시판](#)[로그아웃](#)

예약검색

병원/기관 명 : ## 내과 000 의사

예약 일시 ▼

예약 시간 ▼

예약자 명

전화번호

검색

예약 일시	예약 시간	예약자 명	전화번호	주민 번호	접종 차수	특이사항
2021-10-28	2시	홍OO	010-0000-0000	950000-111111	1차	꿀절 입원 중
2021-10-28	2시	김OO	010-0000-0000	961223-132323	2차	
2021-10-28	2시	이OO	010-0000-0000	931205-1987654	2차	알레르기
2021-10-28	3시	박OO	010-0000-0000	950313-1765413	2차	
2021-10-28	3시	최OO	010-0000-0000	950516-1543636	2차	
2021-10-28	3시	강##	010-0000-0000	950227-1489484	2차	
2021-10-29	10시	김**	010-0000-0000	950321-1877844	1차	
2021-10-29	10시	이@@	010-0000-0000	950403-1878355	1차	
2021-10-29	10시	김##	010-0000-0000	950423-1435665	1차	
2021-10-29	10시	박\$\$	010-0000-0000	950623-1455465	1차	
2021-10-29	10시	최@@	010-0000-0000	950927-1546554	2차	
2021-10-30	11시	이\$\$	010-0000-0000	951112-1354325	2차	
2021-10-30	11시	김@@	010-0000-0000			
2021-10-30	11시	한OO	010-0000-0000			

[Begin](#)[Prev](#)[1](#)[2](#)[3](#)

[홈페이지 로고](#)[예약검색](#)[접종완료자 현황](#)[공지사항](#)[게시판](#)[로그아웃](#)

예약검색

병원/기관 명 : ## 내과 000 의사

예약 일시 ▼

예약 시간 ▼

예약자 명

전화번호

검색

예약 일시	예약 시간	예약자 명	전화번호	주민 번호	접종 차수	특이사항
2021-10-28	2시	홍OO	010-0000-0000	950000-111111	1차	꿀절 입원 중
2021-10-28	2시	김OO	010-0000-0000	961223-132323	2차	
2021-10-28	2시	이OO	010-0000-0000	931205-1987654	2차	알레르기
2021-10-28	3시	박OO	010-0000-0000	950313-1765413	2차	
2021-10-28	3시	최OO	010-0000-0000	950516-1543636	2차	
2021-10-28	3시	강##	010-0000-0000	950227-1489484	2차	
2021-10-29	10시	김**	010-0000-0000	950321-1877844	1차	
2021-10-29	10시	이@@	010-0000-0000	950403-1878355	1차	
2021-10-29	10시	김##	010-0000-0000	950423-1435665	1차	
2021-10-29	10시	박\$\$	010-0000-0000	950623-1455465	1차	
2021-10-29	10시	최@@	010-0000-0000	950927-1546554	2차	
2021-10-30	11시	이\$\$	010-0000-0000	951112-1354325	2차	
2021-10-30	11시	김@@	010-0000-0000			
2021-10-30	11시	한OO	010-0000-0000			

[Begin](#)[Prev](#)[1](#)[2](#)[3](#)

예약 확인

예약 일시

2021-10-29

예약 시간

10시

예약자 명

김##

전화번호

010-0000-0000

주민 번호

950423-1435665

접종 차수

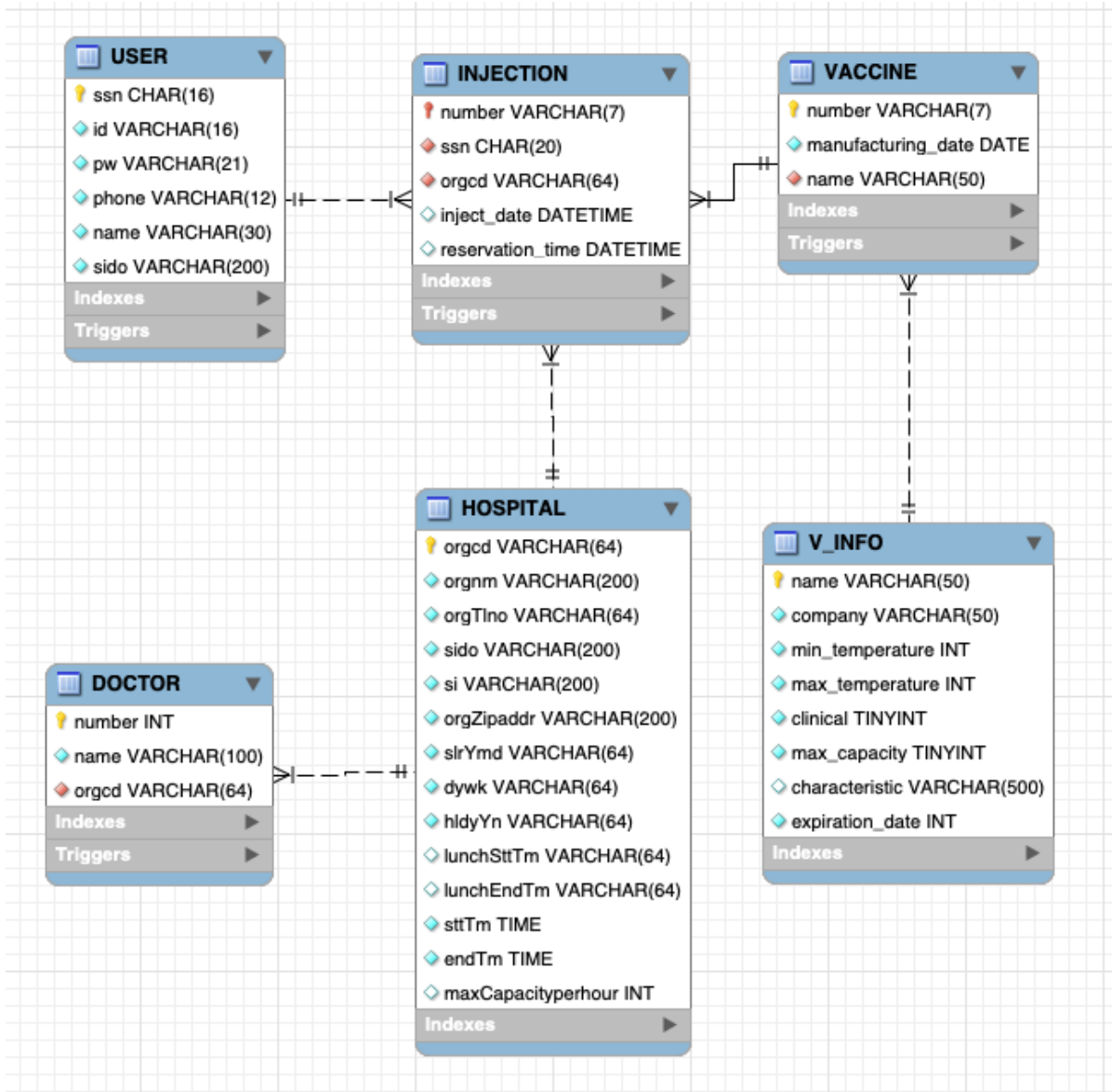
1차

특이사항

[예약 변경](#)[예약 취소](#)[취소](#)

[Begin](#)[Prev](#)[1](#)[2](#)[3](#)[4](#)[.....](#)[10](#)[11](#)[Next](#)[End](#)

■ ER diagram



위는 이번 프로젝트에서 구현한 백신 접종 프로그램 DB의 ER-diagram이다. 먼저 USER, INJECTION, VACCINE, V_INFO, HOSPITAL, DOCTOR의 총 6개의 테이블이 존재한다. 또한 4개의 점선관계(USER - INJECTION, VACCINE - V_INFO, INJECTION - HOSPITAL, HOSPITAL - DOCTOR)와 1개의 실선관계(INJECTION - VACCINE)이 존재하며 실선관계는 부모 테이블의 PK가 자식 테이블의 KEY attribute가 되는 관계, 점선관계는 부모 테이블의 PK가 자식 테이블의 General attribute가 되는 관계이다. USER이 여러 번의 동일 백신 접종이 가능하므로 USER-INJECTION은 1 : N관계, 백신 1병당 접종 가능한 횟수가 여러 번이므로 INJECTION-VACCINE은 N : 1관계, 여러 병의 백신은 동일 백신마다 동일한 백신상세정보가 존재하므로 VACCINE-V_INFO는 N : 1관계, 병원 여러 번 다른 환자들에게 백신을 접종하므로 INJECTION-HOSPITAL은 N : 1관계, 1개의 병원에 여러 명의 의사가 존재하므로 HOSPITAL-DOCTOR은 1 : N의 관계를 가진다.

■ CREATE TABLE Description

V_INFO CREATE TABLE

```
CREATE TABLE V_INFO(  
  name VARCHAR(50) NOT NULL PRIMARY KEY,           #백신명  
  company VARCHAR(50) NOT NULL,                     #회사명  
  min_temperature INT NOT NULL,                     #최소보관온도  
  max_temperature INT NOT NULL,                     #최대보관온도  
  clinical TINYINT NOT NULL,                        #임상3상  
  max_capacity TINYINT NOT NULL,                    #한병당 접종가능인원  
  characteristic VARCHAR(500),                     #특징  
  expiration_date INT UNSIGNED NOT NULL             #유통기한(일)  
);
```

V_INFO TABLE COMPONENT

Column name	Data type	Detailed option	Column Description
name	VARCHAR(50)	NOT NULL, PRIMARY KEY	백신 이름
company	VARCHAR(50)	NOT NULL	제약회사명
min_temperature	INT	NOT NULL	최소 보관온도
max_temperature	INT	NOT NULL	최대 보관온도
clinical	TINYINT	NOT NULL	임상3상 통과여부
max_capacity	TINYINT	NOT NULL	1병당 접종가능인원
characteristic	VARCHAR(500)		특징
expiration_date	INT	NOT NULL	유통기한(일)

V_INFO TABLE DESCRIPTION

각 백신에 대한 상세정보(백신 이름, 제약회사명, 보관온도 범위, 임상3상 통과여부, 1병당 접종가능인원, 특징, 유통기한)가 담긴 테이블이다.

VACCINE CREATE TABLE

```
CREATE TABLE VACCINE(  
    number VARCHAR(7) NOT NULL PRIMARY KEY,           #일련번호  
    manufacturing_date DATE NOT NULL,                   #제조일  
    name VARCHAR(50) NOT NULL,                          #백신명  
    FOREIGN KEY (name) REFERENCES V_INFO(name)  
);  
DELIMITER $$  
CREATE TRIGGER trig_vaccine_number_check BEFORE INSERT ON VACCINE  
FOR EACH ROW  
BEGIN  
    IF (NEW.number REGEXP '^[A-Z]{2}[0-9]{4}$') = 0 THEN  
        SIGNAL SQLSTATE '20001'  
        SET MESSAGE_TEXT = "백신 일련번호는 차례대로 영어대문자 2자리 숫자 4자리로 구성되어야한다.";  
    END IF;  
END$$
```

VACCINE TABLE COMPONENT

Column name	Data type	Detailed option	Column Description
number	VARCHAR(7)	NOT NULL, PRIMARY KEY	일련번호
manufacturing_date	DATE	NOT NULL	제조일자
name	VARCHAR(50)	NOT NULL, FOREIGN KEY OF V_INFO	백신 이름

VACCINE TABLE DESCRIPTION

1병의 백신에 대한 정보(일련번호, 제조일자, 백신 이름)가 담긴 테이블이다. 백신의 일련번호는 순서대로 영어 2자리, 숫자 4자리인 XXNNNN의 형태로 구성되어야 한다.

USER CREATE TABLE

```
CREATE TABLE USER(  
  ssn CHAR(16) NOT NULL PRIMARY KEY,           #주민번호  
  id VARCHAR(16) NOT NULL,                       #아이디  
  pw VARCHAR(21) NOT NULL,                       #비밀번호  
  phone VARCHAR(12) NOT NULL UNIQUE,            #전화번호  
  name VARCHAR(30) NOT NULL,                    #이름  
  sido VARCHAR(200) NOT NULL                    #사용자 주소 시도  
);  
DELIMITER $$  
CREATE TRIGGER trig_user_check BEFORE INSERT ON USER  
FOR EACH ROW  
BEGIN  
  IF (NEW.ssn REGEXP '[0-9]{6}-[0-9]{7}') = 0 THEN  
    SIGNAL SQLSTATE '10001'  
    SET MESSAGE_TEXT = "주민번호는 총 13자리 입니다.";  
  ELSEIF (NEW.id REGEXP '[A-Za-z0-9]{5,15}') = 0 THEN  
    SIGNAL SQLSTATE '10002'  
    SET MESSAGE_TEXT = "id는 영문이나 숫자고 길이는 5~15자 입니다.";  
  ELSEIF (NEW.pw REGEXP '(?=[A-Za-z])(?=[0-9])[A-Za-z0-9]{8,20}') = 0 THEN  
    SIGNAL SQLSTATE '10003'  
    SET MESSAGE_TEXT = "비밀번호는 8~20자, 최소 하나의 문자 및 하나의 숫자를 포함해야 합니다.";  
  ELSEIF (NEW.phone REGEXP '[0-9]{9,12}') = 0 THEN  
    SIGNAL SQLSTATE '10004'  
    SET MESSAGE_TEXT = "전화번호는 9~11자 입니다.";  
  END IF;  
END$$
```

USER TABLE COMPONENT

Column name	Data type	Detailed option	Column Description
ssn	CHAR(50)	NOT NULL, PRIMARY KEY	사용자 주민번호
id	VARCHAR(16)	NOT NULL	ID
pw	VARCHAR(21)	NOT NULL	PASSWORD
phone	VARCHAR(12)	NOT NULL	전화번호
name	VARCHAR(30)	NOT NULL, UNIQUE	이름
sido	VARCHAR(200)	NOT NULL	사용자 주소

USER TABLE DESCRIPTION

사용자 개인정보(주민번호, ID, 비밀번호, 전화번호, 이름, 주소)가 담긴 테이블이다. 주민번호는 6개 숫자 - 7개 숫자를 포함한 총 14자리의 문자로 구성되어 있어야 한다. ID는 영어와 숫자의 조합으로 5~15의 길이를 가져야 한다. 비밀번호는 영어, 숫자 각각 최소 1개씩 포함된 조합으로 8~20의 길이를 가져야 한다. 전화번호는 9~11길이의 숫자로 이루어져 있어야 한다.

HOSPITAL CREATE TABLE

```
CREATE TABLE HOSPITAL (  
  orgcd VARCHAR(64) NOT NULL PRIMARY KEY,           #기관코드  
  orgnm VARCHAR(200) NOT NULL,                       #기관명  
  orgTlno VARCHAR(64) NOT NULL,                     #기관전화번호  
  sido VARCHAR(200) NOT NULL,                       #시, 도  
  si VARCHAR(200) NOT NULL,                          #시  
  orgZipaddr VARCHAR(200) NOT NULL,                 #기관주소  
  slrYmd VARCHAR(64) NOT NULL,                      #기준일자(현재날짜)  
  dywk VARCHAR(64) NOT NULL,                       #기준일자 요일  
  hldyYn VARCHAR(64) NOT NULL,                     #기준일자 휴무일여부  
  lunchSttTm VARCHAR(64),                          #기준일자 점심시작시간  
  lunchEndTm VARCHAR(64),                          #기준일자 점심종료시간  
  sttTm TIME default "09:00:00" NOT NULL,          #기준일자 오픈시간  
  endTm TIME default "18:00:00" NOT NULL,          #기준일자 마감시간  
  maxCapacityperhour INT default 0                 #시간당 접종가능인원수  
);
```

HOSPITAL TABLE COMPONENT

Column name	Data type	Detailed option	Column Description
orgcd	VARCHAR(64)	NOT NULL, PRIMARY KEY	기관코드
orgnm	VARCHAR(200)	NOT NULL	기관명
orgTlno	VARCHAR(64)	NOT NULL	기관전화번호
sido	VARCHAR(200)	NOT NULL	병원주소 (시, 도)
si	VARCHAR(200)	NOT NULL	병원주소 (시)
orgZipaddr	VARCHAR(200)	NOT NULL	병원 상세주소
slrYmd	VARCHAR(64)	NOT NULL	기준일자(현재날짜)
dywk	VARCHAR(64)	NOT NULL	기준일자 요일
hldyYn	VARCHAR(64)	NOT NULL	기준일자 휴무일여부
LunchSttTm	VARCHAR(64)		기준일자 점심시작시간
lunchEndTm	VARCHAR(64)		기준일자 점심종료시간
sttTm	TIME	NOT NULL, Default AM 9:00	기준일자 시작시간
endTm	TIME	NOT NULL, Default PM 6:00	기준일자 종료시간
maxCapacityperhour	INT	NOT NULL, Default 0	시간당 접종가능인원수

HOSPITAL TABLE DESCRIPTION

병원 정보(병원코드, 병원명, 병원전화번호, 병원주소(시,도), 병원주소(시), 병원상세주소, 일자, 요일, 휴무 여부, 점심시간, 운영시간, 시간당 접종가능인원수)가 담긴 테이블이다. 일자, 요일, 휴무여부, 점심시간, 운영시간은 사용자가 접속한 시간을 기준으로 한다.

INJECTION CREATE TABLE

```
CREATE TABLE INJECTION (  
    number VARCHAR(7) PRIMARY KEY NOT NULL,           #일련번호  
    ssn CHAR(20) NOT NULL,                             #접종자 주민번호  
    orgcd VARCHAR(64) NOT NULL,                        #병원테이블 id  
    inject_date datetime DEFAULT NULL,                 #접종날짜  
    reservation_time datetime,                        #예약시간  
    FOREIGN KEY (orgcd) REFERENCES HOSPITAL(orgcd),  
    FOREIGN KEY (ssn) REFERENCES USER(ssn),  
    FOREIGN KEY (number) REFERENCES VACCINE(number)  
);  
DELIMITER $$  
CREATE TRIGGER vaccine_count_check BEFORE INSERT ON INJECTION  
FOR EACH ROW  
BEGIN  
    IF ( EXISTS(  
        SELECT ssn  
        FROM INJECTION  
        WHERE ssn = NEW.ssn  
        GROUP BY ssn  
        HAVING COUNT(*)>1)  
    ) = 0 THEN  
        SIGNAL SQLSTATE '30001'  
        SET MESSAGE_TEXT = "백신접종은 1인당 2번이 최대이다.";  
    END IF;  
END$$
```

INJECTION TABLE COMPONENT

Column name	Data type	Detailed option	Column Description
number	VARCHAR(7)	NOT NULL, PRIMARY KEY, FOREIGN KEY OF VACCINE	일련번호
ssn	CHAR(20)	NOT NULL, FOREIGN KEY OF USER	접종인 주민번호
orgcd	VARCHAR(64)	NOT NULL, FOREIGN KEY OF HOSPITAL	기관코드
inject_date	DATETIME	Default NULL	접종일자
reservation_time	DATETIME		예약시간

INJECTION TABLE DESCRIPTION

접종 관련 정보(백신 일련번호, 접종인 주민번호, 기관코드, 접종일, 예약시간)가 담긴 테이블이다. 백신 접종은 인당 2회(1차, 2차)까지만 접종이 가능하다.

DOCTOR CREATE TABLE

```
CREATE TABLE DOCTOR (  
    number INT PRIMARY KEY NOT NULL,           #의사면허번호  
    name varchar(100) NOT NULL,                #의사 이름  
    orgcd VARCHAR(64) NOT NULL,                 #병원테이블 id  
    FOREIGN KEY (orgcd) REFERENCES HOSPITAL(orgcd)  
);  
DELIMITER $$  
CREATE TRIGGER trig_doctor_number_check BEFORE INSERT ON DOCTOR  
FOR EACH ROW  
BEGIN  
    IF (NEW.number REGEXP '^[0-9]{8}$') = 0 THEN  
        SIGNAL SQLSTATE '40001'  
        SET MESSAGE_TEXT = "의사 면허번호는 차례대로 숫자8자리로 구성되어야한다.";  
    END IF;  
END$$
```

DOCTOR TABLE COMPONENT

Column name	Data type	Detailed option	Column Description
number	INT	NOT NULL, PRIMARY KEY	의사면허번호
name	VARCHAR(100)	NOT NULL	의사 이름
orgcd	VARCHAR(64)	NOT NULL, FOREIGN KEY OF HOSPITAL	기관코드

DOCTOR TABLE DESCRIPTION

의사 관련 정보(의사면허번호, 의사 이름, 병원코드)가 담긴 테이블이다. 의사 면허번호는 8 자리의 숫자로 구성되어 있어야 한다.

■ Type of SQL query used in project

구분	SQL문 Type	개수
Data Definition	CREATE, CREATE USER, ALTER, DROP USER	4
Data Manipulation	INSERT, UPDATE, DELETE, TRIGGER	4
Data Query	SELECT, NESTED SELECT	2
Data Control	REVOKE, GRANT	2
Aggregation	GROUP BY, HAVING	2
Join	JOIN, NATURAL JOIN, LEFT OUTER JOIN	3
Date	TIMESTAMPDIFF, DATE_ADD, DATE_SUB, DATE(MO, H, MIN), NOW	5
View	VIEW	1
String operation	REGEXP(),	1
Transaction	IF-ELSEIF, LIKE, COUNT, FLUSH,	4
Operator	IS NOT NULL, AND, BETWEEN, IN	4
Sorting	ORDER BY	1
etc	SET @@SQL_MODE, AS, DUAL, EXISTS, NOT EXISTS, LIMIT, OFFSET, EXPLAIN, DISTINCT, DELIMITER, SHOW	11
total		44

■ SQL query description

SQL문

```
INSERT USER(ssn, id, pw, phone, name, sido)
```

```
VALUES
```

```
('971220-1111111', 'tester1', 'imtester1', '01073037361', '홍길동', '서울특별시'),  
( '971220-1111112', 'tester2', 'imtester2', '01073037362', '조정우', '서울특별시'),  
( '971220-1111113', 'tester3', 'imtester3', '01073037363', '임경택', '서울특별시'),  
( '971220-1111114', 'tester4', 'imtester4', '01073037364', '류화랑', '서울특별시'),  
( '971220-1111115', 'tester5', 'imtester5', '01073037365', '강성운', '서울특별시'),  
( '971220-1111116', 'tester6', 'imtester6', '01073037366', '서유상', '서울특별시'),  
( '971220-1111117', 'tester7', 'imtester7', '01073037367', '전승권', '서울특별시'),  
( '971220-1111118', 'tester8', 'imtester8', '01073037368', '김하영', '서울특별시'),  
( '971220-1111119', 'tester9', 'imtester9', '01073037369', '이민석', '서울특별시');
```

SQL문 종류

- INSERT

SQL문 설명

INSERT를 사용하여 USER 테이블에 사용자를 추가하는 예제이다.

SQL문 사용위치

사용자 DB구축에 사용

SQL문 실행결과

	ssn	id	pw	phone	name	sido
▶	971220-1111111	tester1	imtester1	01073037361	홍길동	서울특별시
	971220-1111112	tester2	imtester2	01073037362	조정우	서울특별시
	971220-1111113	tester3	imtester3	01073037363	임경택	서울특별시
	971220-1111114	tester4	imtester4	01073037364	류화랑	서울특별시
	971220-1111115	tester5	imtester5	01073037365	강성운	서울특별시
	971220-1111116	tester6	imtester6	01073037366	서유상	서울특별시
	971220-1111117	tester7	imtester7	01073037367	전승권	서울특별시
	971220-1111118	tester8	imtester8	01073037368	김하영	서울특별시
	971220-1111119	tester9	imtester9	01073037369	이민석	서울특별시
✱	NULL	NULL	NULL	NULL	NULL	NULL

SQL문

```
INSERT V_INFO(name, company, min_temperature, max_temperature, clinical, max_capacity, characteristic, expiration_date)
VALUES
('화이자', '(주)화이자만문회사', 30, 50, 3, 10, '화이자는 팡 좋아요.! 맛고 살아남으시길!', 14),
('모더나', '(주)모더나만문회사', 20, 40, 2, 15, '모더나는 팡 좋아요.! 맛고 살아남으시길!', 7),
('아스트라제네카', '(주)아스트라제네카만문회사', 0, 20, 3, 8, '아스트라제네카는 팡 좋아요.! 맛고 살아남으시길!', 21),
('얀센', '(주)얀센만문회사', 10, 30, 3, 15, '얀센은 팡 좋아요.! 맛고 살아남으시길!', 14);
```

SQL문 종류

- INSERT

SQL문 설명

INSERT를 사용하여 V_INFO 테이블에 백신 상세정보를 추가하는 예제이다.

SQL문 사용위치

백신 상세정보 DB구축에 사용

SQL문 실행결과

[illegible]

SQL문

```
INSERT VACCINE(number, manufacturing_date, name)
VALUES
('HE0000', '2021-10-01', '화이자'),
('HE0001', '2021-10-01', '화이자'),
('HE0002', '2021-10-27', '화이자'),
('HE0003', '2021-10-27', '화이자'),
('MO0000', '2021-10-01', '모더나'),
('MO0001', '2021-10-27', '모더나'),
('MO0002', '2021-10-27', '모더나'),
('AE0000', '2021-10-01', '아스트라제네카'),
('AE0001', '2021-10-01', '아스트라제네카'),
('AE0002', '2021-10-23', '아스트라제네카'),
('AE0003', '2021-10-23', '아스트라제네카'),
('AE0004', '2021-10-22', '아스트라제네카'),
('AE0005', '2021-10-01', '아스트라제네카'),
('YA0000', '2021-10-14', '얀센'),
('YA0001', '2020-10-26', '얀센'),
('YA0002', '2020-10-27', '얀센');
```

SQL문 종류

- INSERT

SQL문 설명

INSERT를 사용하여 VACCINE 각 병의 백신 정보를 추가하는 예제이다.

SQL문 사용위치

각 병의 백신 정보 DB구축에 사용

SQL문 실행결과

	number	manufacturing_date	name
▶	AE0000	2021-10-01	아스트라제네카
	AE0001	2021-10-01	아스트라제네카
	AE0002	2021-10-23	아스트라제네카
	AE0003	2021-10-23	아스트라제네카
	AE0004	2021-10-22	아스트라제네카
	AE0005	2021-10-01	아스트라제네카
	HE0000	2021-10-01	화이자
	HE0001	2021-10-01	화이자
	HE0002	2021-10-27	화이자
	HE0003	2021-10-27	화이자
	MO0000	2021-10-01	모더나
	MO0001	2021-10-27	모더나
	MO0002	2021-10-27	모더나
	YA0000	2021-10-14	얀센
	YA0001	2020-10-26	얀센
	YA0002	2020-10-27	얀센
*	NULL	NULL	NULL

SQL문

```
INSERT DOCTOR(number, name, orgcd)
VALUES
(11111111, '안철수', '21303568'),
(22222222, '김상은', '35323175'),
(33333333, '돌팔이', '21345970'),
(44444444, '김성생', '35306611'),
(55555555, '화타', '33316228'),
(66666666, '김지성', '21311072');
```

SQL문 종류

- INSERT

SQL문 설명

INSERT를 사용하여 DOCTOR 테이블에 의사정보를 추가하는 예제이다.

SQL문 사용위치

의사정보 DB구축에 사용

SQL문 실행결과

	number	name	orgcd
▶	11111111	안철수	21303568
	22222222	김상은	35323175
	33333333	돌팔이	21345970
	44444444	김성생	35306611
	55555555	화타	33316228
	66666666	김지성	21311072
*	NULL	NULL	NULL

SQL문

```
INSERT INJECTION(number, ssn, orgcd, inject_date, reservation_time)
```

```
VALUES
```

```
('HE0000', '971220-1111111', '21303568', '2021-10-27 16:00:10', '2021-10-27 16:00:00'),  
( 'HE0001', '971220-1111112', '35323175', '2021-10-27 16:00:10', '2021-10-27 16:00:00'),  
( 'MO0000', '971220-1111113', '21345970', '2021-10-27 16:00:10', '2021-10-27 16:00:00'),  
( 'MO0001', '971220-1111114', '35306611', '2021-10-27 16:00:10', '2021-10-27 16:00:00'),  
( 'AE0000', '971220-1111115', '33316228', '2021-10-27 16:00:10', '2021-10-27 16:00:00');
```

SQL문 종류

- INSERT

SQL문 설명

INSERT를 사용하여 INJECTION 테이블에 접종정보를 추가하는 예제이다.

SQL문 사용위치

백신 접종 DB구축에 사용

SQL문 실행결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-1111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-1111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-1111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0000	971220-1111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-1111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
*	NULL	NULL	NULL	NULL	NULL

SQL문

```
INSERT INJECTION(number, ssn, orgcd, reservation_time)
SELECT "HE0003", "971220-1111119", "21303568", "2021-11-04 14:00:00"
FROM DUAL
WHERE NOT EXISTS(SELECT *
FROM VACCINE AS V JOIN V_INFO AS V_I ON V.name = V_I.name
WHERE V.number = "HE0003" AND DATE_ADD(V.manufacturing_date, INTERVAL V_I.expiration_date DAY) < now())
```

SQL문 종류

- INSERT
- DUAL
- NOT EXISTS
- AS
- JOIN
- DATE_ADD
- NOW
- NESTED SELECT

SQL문 설명

INSERT를 사용하여 사용자가 예약한 백신 접종 정보를 추가하는 예제이다. NOT EXISTS를 사용하여 백신 예약 시 유효기간이 지난 백신은 예약이 불가능하도록 한다.

SQL문 사용위치

백신 예약시 사용

SQL문 실행결과

SQL문 실행 전 결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-1111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-1111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-1111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0000	971220-1111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-1111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
★	NULL	NULL	NULL	NULL	NULL

SQL문 실행 후 결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-1111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-1111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-1111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0003	971220-1111119	21303568	NULL	2021-11-04 14:00:00
	MO0000	971220-1111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-1111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
★	NULL	NULL	NULL	NULL	NULL

SQL문

```
DELETE FROM INJECTION  
WHERE number="HE0003";
```

SQL문 종류

- DELETE

SQL문 설명

DELETE를 사용하여 사용자의 백신 예약을 취소하는 예제이다.

SQL문 사용위치

백신 예약 취소 시 사용

SQL문 실행결과

SQL문 실행 전 결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-11111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-11111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-11111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0003	971220-11111119	21303568	NULL	2021-11-04 14:00:00
	MO0000	971220-11111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-11111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
*	NULL	NULL	NULL	NULL	NULL

SQL문 실행 후 결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-11111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-11111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-11111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0000	971220-11111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-11111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
*	NULL	NULL	NULL	NULL	NULL

SQL문

```
create user woonyDBA;  
grant all privileges on *.* to woonyDBA;  
flush privileges;
```

SQL문 종류

- CREATE

- GRANT

- FLUSH

SQL문 설명

woonyDBA라는 관리자를 추가하고 관리자에게 모든 권한을 부여하는 SQL문이다.

SQL문 사용위치

DB USER정보 추가 시 사용

SQL문 실행결과

SQL문 실행 전 결과

	user	host
▶	mysql.infoschema	localhost
	mysql.session	localhost
	mysql.sys	localhost
	root	localhost

SQL문 실행 후 결과

	user	host
▶	woonyDBA	%
	mysql.infoschema	localhost
	mysql.session	localhost
	mysql.sys	localhost
	root	localhost

SQL문

```
CREATE VIEW RESERVE_BY_TIME_AND_HOSPITAL AS
SELECT orgcd, DATE(reservation_time), HOUR(reservation_time), COUNT(*)
FROM INJECTION
GROUP BY orgcd, DATE(reservation_time), HOUR(reservation_time);
```

SQL문 종류

- CREATE
- VIEW
- AS
- DATE
- COUNT
- GROUP BY

SQL문 설명

CREATW VIEW를 사용하여 INJECTION 테이블에서 해당 병원의 시간별 백신 접종 예약 횟수를 가져와 VIEW를 생성하는 SQL문이다.

SQL문 사용위치

병원마다 시간별 예약 수

SQL문 실행결과

SQL문 실행 전 결과

	Tables_in_db	Table_type
▶	doctor	BASE TABLE
	hospital	BASE TABLE
	injection	BASE TABLE
	user	BASE TABLE
	v_info	BASE TABLE
	vaccine	BASE TABLE

SQL문 실행 후 결과

	Tables_in_db	Table_type
▶	doctor	BASE TABLE
	hospital	BASE TABLE
	injection	BASE TABLE
	reserve_by_time_and_hospital	VIEW
	user	BASE TABLE
	v_info	BASE TABLE
	vaccine	BASE TABLE

SQL문 실행 후 결과

	orgcd	DATE(reservation_time)	HOUR(reservation_time)	COUNT(*)
▶	33316228	2021-10-27	16	1
	21303568	2021-10-27	16	1
	35323175	2021-10-27	16	1
	21345970	2021-10-27	16	1
	35306611	2021-10-27	16	1

SQL문

```
CREATE VIEW RESERVE_LEFT_BY_TIME_AND_HOSPITAL AS
SELECT orgcd, DATE(reservation_time), HOUR(reservation_time), HOSPITAL.maxCapacityperhour-COUNT(*) AS REMAIN
FROM INJECTION NATURAL JOIN HOSPITAL
GROUP BY orgcd, DATE(reservation_time), HOUR(reservation_time);
```

SQL문 종류

- CREATE
- VIEW
- AS
- DATE
- COUNT
- GROUP BY

SQL문 설명

CREATE VIEW를 사용하여 INJECTION 테이블에서 해당 병원의 시간 별 백신 접종 예약자의 수를 가져와 VIEW를 생성하는 SQL문이다.

SQL문 사용위치

병원마다 시간 별 예약 접종자 수 조회 시 사용

SQL문 실행결과

SQL문 실행 전 결과

	Tables_in_db	Table_type
▶	doctor	BASE TABLE
	hospital	BASE TABLE
	injection	BASE TABLE
	reserve_by_time_and_hospital	VIEW
	user	BASE TABLE
	v_info	BASE TABLE
	vaccine	BASE TABLE

SQL문 실행 후 결과

	Tables_in_db	Table_type
▶	doctor	BASE TABLE
	hospital	BASE TABLE
	injection	BASE TABLE
	reserve_by_time_and_hospital	VIEW
	reserve_left_by_time_and_hospital	VIEW
	user	BASE TABLE
	v_info	BASE TABLE
	vaccine	BASE TABLE

SQL문 실행 후 결과

	orgcd	DATE(reservation_time)	HOUR(reservation_time)	REMAIN
▶	33316228	2021-10-27	16	0
	21303568	2021-10-27	16	0
	35323175	2021-10-27	16	0
	21345970	2021-10-27	16	0
	35306611	2021-10-27	16	0

SQL문

```
ALTER TABLE USER CHANGE name nickname VARCHAR(30);
```

SQL문 종류

- ALTER

SQL문 설명

ALTER를 사용하여 USER 테이블의 name을 nickname으로 변경하는 예제이다.

SQL문 사용위치

TABLE내부 속성 이름 변경 시 사용

SQL문 실행결과

SQL문 실행 전 결과

	ssn	id	pw	phone	name	sido
▶	971220-1111111	tester1	imtester1	01073037361	홍길동	서울특별시
	971220-1111112	tester2	imtester2	01073037362	조정우	서울특별시
	971220-1111113	tester3	imtester3	01073037363	임경택	서울특별시
	971220-1111114	tester4	imtester4	01073037364	류화랑	서울특별시
	971220-1111115	tester5	imtester5	01073037365	강성운	서울특별시
	971220-1111116	tester6	imtester6	01073037366	서유상	서울특별시
	971220-1111117	tester7	imtester7	01073037367	전승권	서울특별시
	971220-1111118	tester8	imtester8	01073037368	김하영	서울특별시
	971220-1111119	tester9	imtester9	01073037369	이민석	서울특별시
★	NULL	NULL	NULL	NULL	NULL	NULL

SQL문 실행 후 결과

	ssn	id	pw	phone	nickname	sido
▶	971220-1111111	tester1	imtester1	01073037361	홍길동	서울특별시
	971220-1111112	tester2	imtester2	01073037362	조정우	서울특별시
	971220-1111113	tester3	imtester3	01073037363	임경택	서울특별시
	971220-1111114	tester4	imtester4	01073037364	류화랑	서울특별시
	971220-1111115	tester5	imtester5	01073037365	강성운	서울특별시
	971220-1111116	tester6	imtester6	01073037366	서유상	서울특별시
	971220-1111117	tester7	imtester7	01073037367	전승권	서울특별시
	971220-1111118	tester8	imtester8	01073037368	김하영	서울특별시
	971220-1111119	tester9	imtester9	01073037369	이민석	서울특별시
★	NULL	NULL	NULL	NULL	NULL	NULL

SQL문

```
SELECT *
FROM INJECTION
WHERE inject_date BETWEEN DATE_SUB(now(), INTERVAL 7 DAY) AND now();
```

SQL문 종류

- BETWEEN - DATE_SUB - NOW

SQL문 설명

SELECT를 사용하여 INJECTION 테이블에서 최근 접종된 백신을 조회하는 예제이다. WHERE절에 DATE BETWEEN을 사용하여 INTERVAL을 부여하고, 조회 기간은 조회 시간을 기준으로 7일 이전 까지 기록을 조회하도록 설정하였다.

SQL문 사용위치

기준 시간으로부터 7일 이내에 접종된 백신 기록 조회 시 사용

SQL문 실행결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-1111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-1111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-1111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0000	971220-1111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-1111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
✱	NULL	NULL	NULL	NULL	NULL

SQL문

```
SELECT H.sido, COUNT(*)  
FROM HOSPITAL H NATURAL JOIN INJECTION I  
GROUP BY H.sido;
```

SQL문 종류

- COUNT

- NATURAL JOIN

- GROUP BY

SQL문 설명

SELECT를 사용하여 HOSPITAL 테이블과 INJECTION 테이블에서 백신 접종현황을 조회하는 예제이다. GROUP BY 주소를 사용하여 지역별로 접종 현황을 조회할 수 있도록 주소로 묶어주었다.

SQL문 사용위치

지역별 접종 현황 조회 시 사용

SQL문 실행결과

	sido	COUNT(*)
▶	부산광역시	2
	충청북도	1
	전라북도	2

SQL문

```
SELECT U.sido, COUNT(inject_date) / COUNT(*) * 100 AS 접종률
FROM USER U LEFT OUTER JOIN INJECTION I ON U.ssn = I.ssn
GROUP BY U.sido
HAVING 접종률 > 50;
```

SQL문 종류

- COUNT
- AS
- LEFT OUTER JOIN
- GROUP BY
- HAVING

SQL문 설명

전체 지역 중 백신 평균 접종률이 50%가 넘는 지역만 SELECT하는 SQL문이다. GROUP BY를 사용하여 지역별로 묶은 뒤 HAVING조건을 통해 50%가 넘는 지역을 선별하도록 하였다.

SQL문 사용위치

지역별 평균 백신 접종률 조회 시 사용

SQL문 실행결과

	sido	접종률
▶	서울특별시	55.5556

SQL문

```
SELECT inject_date, COUNT(*)  
FROM INJECTION  
WHERE TIMESTAMPDIFF(DAY, now(), inject_date) < 7 AND inject_date IS NOT NULL  
GROUP BY inject_date;
```

SQL문 종류

- COUNT
- TIMESTAMPDIFF
- NOW
- AND
- IS NOT NULL
- GROUP BY

SQL문 설명

날짜 별 백신 접종 횟수를 SELECT하는 SQL문이다. TIMESTAMPDIFF를 통해 기준 일자로부터 7일 이전의 data만 불러오도록 하였고, GROUP BY를 통해 동일한 접종일자를 묶었다.

SQL문 사용위치

일별 백신 접종 통계 조회 시 사용

SQL문 실행결과

	inject_date	COUNT(*)
▶	2021-10-27 16:00:10	5

SQL문

```
SELECT MONTH(inject_date) AS m_date, COUNT(*)  
FROM INJECTION  
WHERE TIMESTAMPDIFF(MONTH, now(), inject_date) < 6 AND inject_date IS NOT NULL  
GROUP BY m_date;
```

SQL문 종류

- DATE
- AS
- COUNT
- TIMESTAMPDIFF
- NOW
- AND
- IS NOT NULL
- GROUP BY

SQL문 설명

월별 백신 접종 횟수를 SELECT하는 SQL문이다. TIMESTAMPDIFF를 통해 기준 일자로부터 6개월 이내의 data만 불러오도록 하였고, GROUP BY를 통해 동일한 접종월을 묶었다.

SQL문 사용위치

월별 백신 접종 통계 조회 시 사용

SQL문 실행결과

	m_date	COUNT(*)
▶	10	5

SQL문

```
SELECT orgnm, sido, si
FROM HOSPITAL
WHERE orgnm LIKE "%서울%" AND si = "강남구" AND sido = "서울특별시"
ORDER BY orgnm LIMIT 20 OFFSET 0;
```

SQL문 종류

- LIKE
- AND
- ORDER BY
- LIMIT
- OFFSET

SQL문 설명

서울특별시 강남구에 존재하는 병원 중 병원 이름에 서울이 들어간 병원을 조회하는 SQL문이다.
무한 스크롤 기능을 위해 LIMIT와 OFFSET을 설정하였다.

SQL문 사용위치

지역별 병원 조회 시 사용,

SQL문 실행결과

	orgnm	sido	si
▶	남서울정형외과의원	서울특별시	강남구
	대치서울영재소아청소년과의원	서울특별시	강남구
	삼성서울유외과의원	서울특별시	강남구
	서울ND(엔디)의원	서울특별시	강남구
	서울박내과의원	서울특별시	강남구
	서울배내과의원	서울특별시	강남구
	서울베스트의원	서울특별시	강남구
	서울본브릿지병원	서울특별시	강남구
	서울삼성내과의원	서울특별시	강남구
	서울손내과의원	서울특별시	강남구
	서울수면의원	서울특별시	강남구
	서울연세21세기의원	서울특별시	강남구
	서울특별시강남구보건소	서울특별시	강남구
	재단법인서울의과학연구소 강...	서울특별시	강남구
	재단법인원불교서울교구 강남...	서울특별시	강남구
	청담서울이비인후과의원	서울특별시	강남구

SQL문

```
SELECT *
FROM HOSPITAL
ORDER BY orgnm ASC;
```

SQL문 종류

- ORDER BY

- ASC

SQL문 설명

병원 이름으로 오름차순 정렬을 진행하는 SQL문이다.

SQL문 사용위치

병원 조회 시 사용

SQL문 실행결과

orgcd	orgnm	orgTlno	sido	si	orgZipaddr	slrYmd	dywk	hldyYn
▶ 21303568	(사)대한결핵협회부산광역시지부 복십자의원	051-611-3622	부산광역시	남구	부산광역시 남구 수영로 299, (대연동)	20211030	토요일	N
35323175	(사)대한산업보건협회부산산업보건센터부산서...	063-731-1242	전라북도	군산시	전라북도 군산시 산단남북로 169, (오식도동) 2층	20211030	토요일	N
21345970	(사)대한산업보건협회남부산의원	051-710-6888	부산광역시	강서구	부산광역시 강서구 명지오션시티9로 21, (명지...	20211030	토요일	Y
35306611	(사)대한산업보건협회전북산업보건센터 전북의원	063-225-1242	전라북도	전주시	전라북도 전주시 덕진구 기린대로 1030, (여의동)	20211030	토요일	N
33316228	(사)대한신장복지회 대신의원	043-224-0825	충청북도	청주시	충청북도 청주시 청원구 충정대로 93 지하 1층 ...	20211030	토요일	N
21311072	(사)인구보건복지협회 부산시지회 가족보건의원	051-624-5581	부산광역시	수영구	부산광역시 수영구 수영로 425, (남천동)	20211030	토요일	N
11332468	(사)인구보건복지협회 서울지회 가족보건의원	02-467-8913	서울특별시	광진구	서울특별시 광진구 긴고랑로13길 62, (중곡동) ...	20211030	토요일	N
33302561	(사)인구보건복지협회 충북세종지회 가족보건...	043-270-5900	충청북도	청주시	충청북도 청주시 흥덕구 내수동로 20, (북대동, ...	20211030	토요일	N
31310702	(사)인구보건복지협회경기도지회가족보건의원	031-256-4644	경기도	수원시	경기도 수원시 팔달구 고화로14번길 11, (매산...	20211030	토요일	Y
35303875	(사)인구보건복지협회전북지회가족보건의원	063-240-2345	전라북도	전주시	전라북도 전주시 덕진구 백제대로 720, (인후동...	20211030	토요일	N
34334963	(사)장애우자활추진본부부설충남의원	042-226-9641	대전광역시	동구	대전광역시 동구 대전로 822, (정동) 남복 B/D 3 층	20211030	토요일	N
31317537	(사)한국건강관리협회건강증진의원	031-251-6151	경기도	수원시	경기도 수원시 장안구 경수대로 857, (조원동)	20211030	토요일	Y
35305011	(사)한국건강관리협회건강증진의원	062-259-8900	전라북도	전주시	전라북도 전주시 덕진구 사평로 40, (덕진동1가)	20211030	토요일	Y
36307530	(사단법인)한국건강관리협회건강증진의원	062-363-4040	광주광역시	서구	광주광역시 서구 대남대로 432, (농성동) 한국...	20211030	토요일	Y
34201866	(신)제일병원	041-931-8330	충청남도	보령시	충청남도 보령시 대해로 52, (공촌동)	20211030	토요일	N
38201771	(의) 내경의료재단 울산제일병원	052-220-3300	울산광역시	남구	울산광역시 남구 남산로354번길 26, (신정동)	20211030	토요일	N
11395125	(의) 열린의료재단 혜화열린의원	02-929-2375	서울특별시	성북구	서울특별시 성북구 동소문로 125, (동선동4가) ...	20211030	토요일	N
35280662	(의) 예림의료재단 정복요양병원	063-540-8500	전라북도	김제시	전라북도 김제시 남북로 202, (요촌동)	20211030	토요일	N
11920246	(의)가산의료재단 광동한방병원	02-2222-4888	서울특별시	강남구	서울특별시 강남구 봉은사로 612, (삼성동)	20211030	토요일	N
38203391	(의)경빈의료재단 김해센텀병원	055-330-9100	경상남도	김해시	경상남도 김해시 구지로 56, (내동, 김해센텀병원)	20211030	토요일	N
31101011	(의)나사렛의료재단 나사렛국제병원	032-899-9999	인천광역시	연수구	인천광역시 연수구 연우길로 98-, (동춘동)	20211030	토요일	N
21351821	(의)삼세의료재단 물사랑내과의원	051-413-0055	부산광역시	영도구	부산광역시 영도구 절영로 30, (대교동1가) 삼...	20211030	토요일	N
37201271	(의)서봉의료재단 보강병원	053-641-9200	대구광역시	달서구	대구광역시 달서구 월배로 102, (진천동)	20211030	토요일	N
21201561	(의)서일의료재단기장병원	051-723-0171	부산광역시	기장군	부산광역시 기장군 기장을 대청로72번길 6, 6	20211030	토요일	N
12348287	(의)성광의료재단 차용의원	02-3015-5418	서울특별시	강남구	서울특별시 강남구 도산대로 442, (청담동, 2층...	20211030	토요일	Y
12315613	(의)성광의료재단차용건진의원	02-1800-7750	서울특별시	강남구	서울특별시 강남구 테헤란로 509, (삼성동, 지...	20211030	토요일	N
31100694	(의)성세의료재단 뉴성민병원	032-726-1000	인천광역시	서구	인천광역시 서구 신석로 70 (석남동)	20211030	토요일	N
31211216	(의)성세의료재단 영종국제병원	032-721-3000	인천광역시	중구	인천광역시 중구 하늘별빛로65번길 7-9, (중산...	20211030	토요일	N
31287395	(의)수지의료재단수지요양병원	031-262-0050	경기도	용인시	경기도 용인시 수지구 풍덕천로22번길 15, (풍...	20211030	토요일	N
32290071	(의)아나의료재단 강릉아나병원	033-610-1100	강원도	강릉시	강원도 강릉시 용지로 129-0, (옥천동)	20211030	토요일	N
12336491	(의)열린의료재단 예인의원	02-3661-7582	서울특별시	강서구	서울특별시 강서구 양천로 366, (가양동, 미라...	20211030	토요일	N
41332172	(의)열린의료재단광주열린의원	031-767-7585	경기도	광주시	경기도 광주시 중앙로 118, (경안동) 3,4,5층	20211030	토요일	Y
31100953	(의)영문의료재단 다보스병원	031-8021-2114	경기도	용인시	경기도 용인시 처인구 백옥대로1082번길 18, (...	20211030	토요일	N
31282296	(의)유일재단 하나에요양병원	031-339-1000	경기도	용인시	경기도 용인시 처인구 백옥대로916번길 48, (남...	20211030	토요일	N
38201372	(의)인석의료재단 보람병원	052-278-0114	울산광역시	남구	울산광역시 남구 돌길로336번길 10, (삼산동)	20211030	토요일	N
31365922	(의)인월의료재단인천제일의원	032-502-6002	인천광역시	부평구	인천광역시 부평구 일신로 8, (일신동) 지1층, 1...	20211030	토요일	N
31287948	(의)현당의료재단 불불요양병원	031-659-7800	경기도	용인시	경기도 용인시 수지구 포은대로59번길 17, (상...	20211030	토요일	N
31204112	(의)효상의료재단 용인서울병원	031-337-0114	경기도	용인시	경기도 용인시 처인구 고림로 81, (고림동)	20211030	토요일	N
38344823	(재)대한불교관음종 대동의원	055-335-1121	경상남도	김해시	경상남도 김해시 대동면 동남로85번길 48, 48	20211030	토요일	N
31204112	(재)대한불교관음종 대동의원	055-335-1121	경상남도	김해시	경상남도 김해시 대동면 동남로85번길 48, 48	20211030	토요일	N

SQL문

```
SELECT DISTINCT company  
FROM V_INFO;
```

SQL문 종류

- DISTINCT

SQL문 설명

SELECT를 사용하여 V_INFO 테이블에 존재하는 백신 상세정보 중 제약회사 이름을 가져오는 SQL문이다. DISTINCT를 사용하여 중복을 제거하였다.

SQL문 사용위치

백신 제조사 리스트 조회 시 사용

SQL문 실행결과

	company
▶	(주)모더나만든회사임
	(주)아스트라제네카만든회사임
	(주)안센만든회사임
	(주)화이자만든회사임

SQL문

```
SELECT *
FROM V_INFO
WHERE name IN(
    SELECT name
    FROM V_INFO
    WHERE name = '화이자')
```

SQL문 종류

- IN
- NESTED SELECT

SQL문 설명

V_INFO 테이블에서 화이자라는 이름을 가진 백신의 상세정보를 모두 가져오는 SQL문이다.

SQL문 사용위치

백신 상세정보 조회 시 사용

SQL문 실행결과

[illegible]

SQL문

```
SELECT I.inject_date, H.orgnm, V.name, U.name
FROM INJECTION I, USER U, HOSPITAL H, VACCINE V
WHERE I.ssn=U.ssn AND I.orgcd=H.orgcd AND I.number=V.number AND U.ssn = "971220-1111115";
```

SQL문 종류

- AS

- AND

SQL문 설명

SELECT를 사용하여 INJECTION, USER, HOSPITAL, VACCINE 테이블을 조회한 뒤 주민번호가 조건과 같은 사용자의 개인별 백신 접종 현황을 조회하는 SQL문이다.

SQL문 사용위치

개인별 백신 접종 현황 조회 시 사용

SQL문 실행결과

	inject_date	orgnm	name	name
▶	2021-10-27 16:00:10	(사)대한신장복지회 대신의원	아스트라제네카	강성운

SQL문

```
SELECT COUNT(*)=1
FROM USER
WHERE ID = "tester1" AND PW = "imtester1";
```

SQL문 종류

- COUNT

- AND

SQL문 설명

입력으로 들어온 ID와 PW가 일치하는 사용자가 존재하는지 확인하는 SQL문이다. 사용자가 존재하면 1, 존재하지 않으면 0을 반환한다.

SQL문 사용위치

로그인 시 사용

SQL문 실행결과

사용자 존재 시 결과

	COUNT(*)=1
▶	1

사용자 존재하지 않을 시 결과

	COUNT(*)=1
▶	0

SQL문

`SHOW TABLES;`

SQL문 종류

- SHOW

SQL문 설명

DB에 존재하는 모든 TABLE들을 보여주는 SQL문이다.

SQL문 사용위치

DB 관리자에 의해 사용됨

SQL문 실행결과

	Tables_in_db
▶	doctor
	hospital
	injection
	reserve_by_time_and_hospital
	reserve_left_by_time_and_hospital
	user
	v_info
	vaccine

SQL문

```
UPDATE INJECTION
SET reservation_time="2020-10-28 18:00:00"
WHERE orgcd = "21303568" AND EXISTS (SELECT *
FROM RESERVE_LEFT_BY_TIME_AND_HOSPITAL
WHERE REMAIN>0);
```

SQL문 종류

- UPDATE
- SET
- EXISTS
- AND

SQL문 설명

INJECTION 테이블에 존재하는 예약 정보를 변경하는 SQL문이다. RESERVE_LEFT_BY_TIME_AND_HOSPITAL VIEW를 이용하여 해당 병원의 백신 접종이 가능한 시간대를 체크하여 해당 시간으로 변경이 가능하면 update를 진행한다.

SQL문 사용위치

사용자 예약시간 변경 시 사용

SQL문 실행결과

SQL문 실행 전 결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-1111115	33316228	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0000	971220-1111111	21303568	2021-10-27 16:00:10	2021-10-27 16:00:00
	HE0001	971220-1111112	35323175	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0000	971220-1111113	21345970	2021-10-27 16:00:10	2021-10-27 16:00:00
	MO0001	971220-1111114	35306611	2021-10-27 16:00:10	2021-10-27 16:00:00
*	NULL	NULL	NULL	NULL	NULL

SQL문 실행 후 결과

	number	ssn	orgcd	inject_date	reservation_time
▶	AE0000	971220-1111115	33316228	2021-10-27 16:00:10	2020-10-27 16:00:00
	HE0000	971220-1111111	21303568	2021-10-27 16:00:10	2020-10-30 18:00:00
	HE0001	971220-1111112	35323175	2021-10-27 16:00:10	2020-10-27 16:00:00
	MO0000	971220-1111113	21345970	2021-10-27 16:00:10	2020-10-27 16:00:00
	MO0001	971220-1111114	35306611	2021-10-27 16:00:10	2020-10-27 16:00:00
*	NULL	NULL	NULL	NULL	NULL

SQL문

```
EXPLAIN(SELECT inject_date, COUNT(*)  
FROM INJECTION  
WHERE TIMESTAMPDIFF(DAY, now(), inject_date) < 7 AND inject_date IS NOT NULL  
GROUP BY inject_date);
```

SQL문 종류

- EXPLAIN
- COUNT
- TIMESTAMPDIFF
- DATE
- NOW
- AND
- IS NOT NULL
- GROUP BY

SQL문 설명

MySQL 쿼리 optimizer가 해당 SQL문을 실행할 때 세우는 최적의 계획을 확인하는 SQL문이다.
이를 통해 DBA가 해당 SQL문을 어떻게 최적화할지에 대해 insight를 제공하게 된다.

SQL문 사용위치

DBA가 DB관리시 사용

SQL문 실행결과

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
▶	1	SIMPLE	INJECTION	NULL	ALL	NULL	NULL	NULL	NULL	5	80.00	Using where; Using temporary

SQL문

```
revoke DELETE, DROP on *.* FROM woonyDBA;  
flush privileges;
```

SQL문 종류

- REVOKE
- FLUSH

SQL문 설명

REVOKE를 사용하여 DBA USER의 DELETE, DROP권한을 취소하는 SQL문이다.

SQL문 사용위치

DBA 권한 취소에 사용

SQL문 실행결과

SQL문 실행 전 결과

	Grants for woonyDBA@%
▶	GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER...
	GRANT APPLICATION_PASSWORD_ADMIN,AUDIT_ADMIN,BACKUP_ADMIN,BINLOG_ADMIN,BINLOG_ENCRYPTION_ADMIN,CLONE_ADMIN,CONNECTION_...

SQL문 실행 후 결과

	Grants for woonyDBA@%
▶	GRANT SELECT, INSERT, UPDATE, CREATE, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER, CREATE TE...
	GRANT APPLICATION_PASSWORD_ADMIN,AUDIT_ADMIN,BACKUP_ADMIN,BINLOG_ADMIN,BINLOG_ENCRYPTION_ADMIN,CLONE_ADMIN,CONNECTION_...