내컴퓨터 -> 속성 -> 고급시스템 설정-> 환경변수 -> 시스템변수 -> 변수에 Path 입력과 mysql 경로를 값에 입력

-> 미설정시 cmd 창에서 순서대로 mysql 폴더 경로로 접근해야 한다

cmd

mysql -u root -p

비밀번호 입력

show databases; -> 데이터베이스 목록 표시

use 데이터베이스명; -> 데이터베이스 선택

show tables; -> 테이블 목록 표시

use mysql -> mysql 실행

show grants for ‘사용자명’@’호스트명’; -> 사용자 권한 확인

create user ‘사용자명’@localhost identified by ‘암호’; -> 사용자 생성

delete from user where user=’사용자명’; -> 사용자 삭제

select user, host from user; -> 사용자 목록 확인

show variables like “%version%”; -> mysql 버전 확인

mysql -E -u root -p -> 쿼리 결과를 세로로 보기

status; -> 현재 상태 확인

use 테이블명; -> 해당 테이블로 이동

explain 테이블명; -> 해당 테이블의 속성 정보 확인

DDL - 데이터 정의어

create - 데이터베이스, 테이블 생성

alter - 데이터베이스, 테이블 수정

drop - 데이터베이스, 테이블 삭제

truncate - 데이터베이스, 테이블을 초기화

데이터베이스 생성/삭제

create database test123; -> test123 데이터베이스를 생성

drop database test123; ->test123 데이터베이스를 삭제

DML - 데이터 조작어(데이터를 관리)

테이블 생성/삭제

create table test1(no int, name varchar(10)); -> test1 테이블에 변수와 자료형을 선언해서 생성

explain test1; -> test1 테이블의 속성 정보 확인

create table 테이블명(컬럼명1, 데이터타입, [default 형식], 컬럼명2, 데이터타입, [default 형식],

unique index 컬럼명\_unique(컬럼명 asc), -> 유니크키 지정(null 입력 가능)

constraint 테이블명\_PK primary key(컬럼명), -> 기본키 지정(not null)

constraint 테이블명\_FK foreign key(컬럼명) references 상대테이블명(기본키), ->외래키 지정

constraint chk\_테이블명 check(논리식)

);

외래키를 제외한 생성 예시

create table test(no int not null, no2 int, no3 int,

unique index no2\_unique(no2 asc),

constraint test\_pk primary key(no),

constraint chk\_test check(no>=1));

테이블 수정(테이블 생성 후 사용)

alter table 테이블명

add 컬럼명 데이터타입;

컬럼 삭제

alter table 테이블명

drop 컬럼명;

컬럼 수정

alter table 테이블명

modify 컬럼명 데이터타입;

컬럼명 수정

alter table 테이블명

change 기존컬럼명 바꿀컬럼명 데이터타입;

제약조건 추가

alter table 테이블명

add constraint 제약조건명 제약조건(컬럼명);

제약조건 삭제

alter table 테이블명

drop constraint 제약조건명;

테이블 초기화

truncate table 테이블명;

테이블 삭제

drop table 테이블명;

delete - 데이터는 지워지지만 테이블 용량은 줄어들지 않고 잘못 삭제한 경우 되돌릴 수 있다

원하는 데이터만 선택해서 지울 수 있다

truncate - 삭제 후 용량이 줄어들며 되돌릴 수 없다

drop - 테이블 전체를 삭제하고 되돌릴 수 없다

예제1-1) 데이터베이스 green과 테이블 subject를 만들어서 속성을 추가

create table subject(no int not null, title varchar(20), time int,

unique index no\_unique(no asc),

constraint subject\_pk primary key(no),

constraint chk\_subject check(no>=1));

insert into 테이블명 values(3, '홍길동', 3'); -> 타입의 순서를 안다면 컬럼값만 입력해줘도 속성 추가 가능

예제1-2) student 테이블 생성

no int, name varchar(11), age int

예제 1-3) subject 테이블에서 no를 기본키로 지정

alter table subject

add constraint subject\_pk primary key(no);

예제 1-4) student 테이블에서 no를 기본키로 지정

alter table student

add constraint student\_pk primary key(no);

예제 1-5) class 테이블 생성

no int, subject\_no int, teacher varchar(11), start\_date datetime, end\_date datetime

예제 1-6) class 테이블의 기본키와 외래키 설정

-> no를 기본키로

alter table class

add constraint class\_pk primary key(no);

-> class 테이블의 subject\_no를 외래키로 지정하고 subject 테이블의 no와 연결

alter table class

add constraint class\_fk foreign key(subject\_no) references subject(no);

DCL - 데이터 제어어(테이블의 권한 제어)

GRANT 특정 데이터베이스 사용자에게 특정 작업에 대한 수행 권한 부여

-> GRANT 명령어 ON 데이터베이스명.테이블 TO '아이디'@'localhost';

REVOKE 특정 데이터베이스 사용자에게 특정 작업에 대한 수행 권한을 박탈, 회수

-> REVOKE 명령어 ON 데이터베이스명.테이블 FROM '아이디'@'localhost';

COMMIT 작업한 결과를 물리적 디스크로 저장하고, 작업이 정상적으로 완료됨을 관리자에게 알려주는 명령어

ROLLBACK 작업했던 내용을 원래의 상태로 복구하는 명령어

INSERT, UPDATE, DELETE와 같은 트랜잭션의 작업을 취소

COMMIT 명령어를 사용하기 이전의 상태만 ROLLBACK 가능

예제 2-1) greenmaster 사용자를 추가하고 green DB 모든 권한을 부여

select user, host from user;

create user 'greenmaster'@localhost identified by 'cjgreen';

select user, host from user;

grants all on green.\* to 'greenmaster'@'localhost';

show grants for 'greenmaster'@'localhost';

예제 2-2) green 사용자를 추가하고 green DB에서 데이터를 관리할 수 있는 권한을 부여

select user, host from user;

create user 'green'@localhost identified by 'cjgreen';

select user, host from user;

grant insert, select, update, delete on green.\* to 'green'@'localhost';

show grants for 'green'@'localhost';