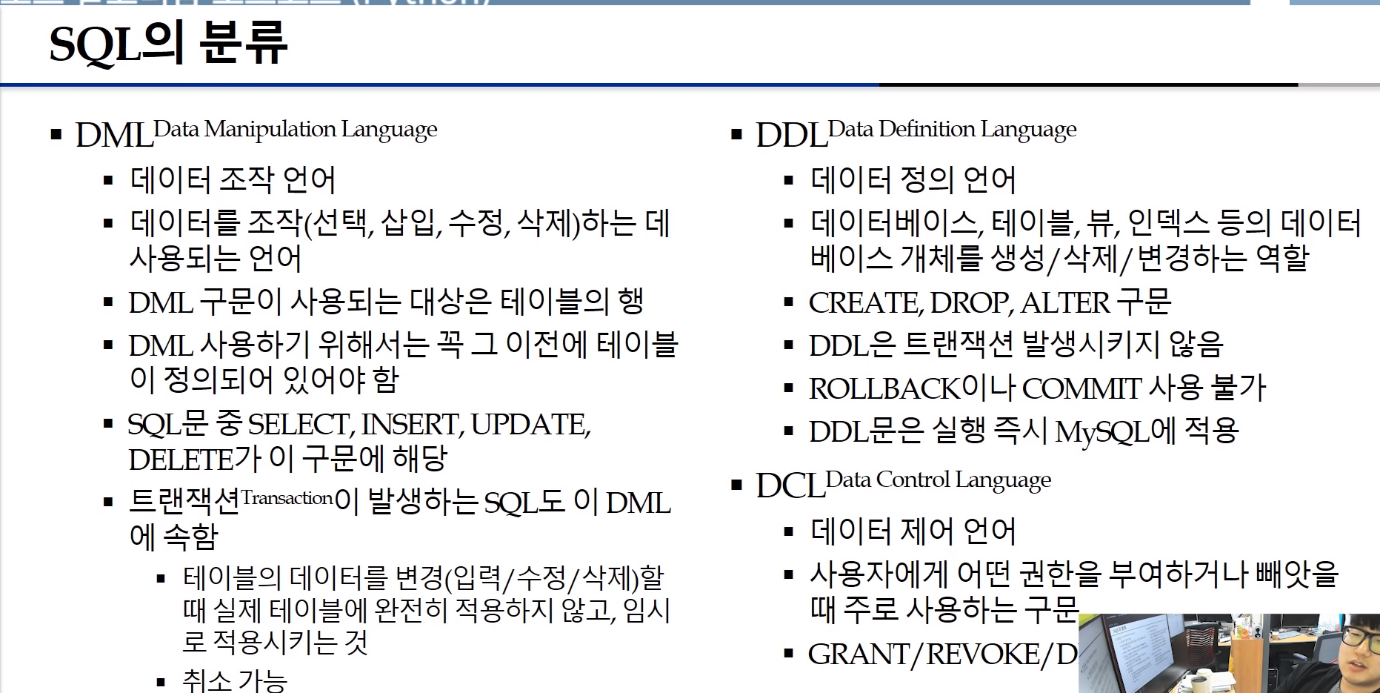
MYSQL

* 데이터 베이스 관리 시스템



**SHOW** DATABASES – 데이터베이스 보기

**USE** WORLD - WORLD라는 데이터 베이스 사용

**SHOW** TABLES – 테이블 보여줘

**SHOW TABLE STATUS**; - 테이블 현재 상태

**DESCRIBE** CITY - CITY테이블의 정보(**DESC**)

**SELECT**~**FROM** CITY – CITY 테이블의 모든(\*)전체를 보여줌(~에 NAME같은 거 쓸수 있음)

SELECT~FROM CITY **WHERE** – 특정 조건만 출력(AND와 같은 연산자 사용가능)

SELECT~FROM CITY WHERE ~ **BETWEEN** 100 **AND** 200; - 100과 200사이에 있는것만

SELECT~FROM CITY WHERE NAME **IN**(‘SEOUL’,NEW YORK’) – 네임중에 서울과 뉴욕만 보고 싶을때

SELECT~FROM CITY WHERE ~ **LIKE** ‘**KO\_**’ – 문자열 검색(KO뒤에 문자 하나오는 거 검색)

‘**TEL %**’ – TEL로 시작하는 거 모두 탐색

SELECT **COUNT**(ANIMAL\_ID) FROM ANIMAL\_INS – ID 몇 개인지 세기

**서브쿼리**(쿼리안에 쿼리가 또 들어가 있는 것) – 질문을 두번한다고 생각

SELECT \* FROM CITY WHERE CODE = ( **SELECT CODE FROM CITY WHERE NAME = ‘SEOUL’**);

* 서울이라는 이름의 도시의 코드를 찾고 그 코드를 가진 도시모두 출력

SELECT \* FROM CITY WHERE POPULATION > **ANY** ( SELECT POPULATION FROM CITY WHERE NAME = ‘SEOUL’);

**ANY** – 하나만 만족해도됨 , **ALL** – 모두 만족해야함

SELECT \* FROM CITY **ORDER BY** POPULATION **DESC** – DESC(큰거부터) 없으면 기본 정렬

SELECT **DISTINCT** CODE FROM CITY; - 중복된거 제외하고 보여줌

SELECT \* FROM CITY ORDER BY POPULATION DESC **LIMIT** 10 – 상위 10개만 출력

SELECT CODE , **MAX**(POPPULATION) AS ‘AVER’ FROM CITY **GROUP BY** CODE – 같은 코드들을 하나로 묶어서 코드와 인구수 가장 큰거를 보여줘(컬럼 명이 AVER이 됨)

SELECT CODE , MAX(POPPULATION) AS ‘AVER’ FROM CITY GROUP BY CODE **HAVING** MAX(POPPULATION) > 200 – HAVING은 GROUP BY다음에 조건을 걸어줄 때 나옴

SELECT CODE , NAME, SUM(POPPULATION) FROM CITY GROUP BY CODE, NAME **WITH ROLLUP –**코드가 같은거를 모두 보여주고 마지막에 합계를 보여줌(GROUP BY와 사용해야함

SELECT \* FROM CITY **JOIN** COUNTRY **ON** CITY.CountryCode = COUNTRY.Code - CITY테이블과 COUNTRY테이블을 특정 조건에 맞춰 합치고 싶을 때 사용

**IF** (조건문, 참일때, 거짓일때)

**내장함수**

SELECT **LENGTH**(‘ASDAS’) – 길이 출력

SELECT **CONCAT**(‘ASDAS’,’WWW’) – 문자열 합치기(문자열중 하나라도 NULL이면 NULL반환

SELECT **LOCATE**(‘ABC’, ‘ABCABCABC’) – 어느 위치에 있는지 출력(SQL에서는 인덱스가 1부터 시작)

SELECT **LEFT**(‘ABCDEFGHIGKLMNOP’,5) – 왼쪽에서 5개 출력

SELECT **RIGHT**(‘ABCDEFGHIGKLMNOP’,5) – 오른쪽에서 5개 출력

SELECT **LOWER**(‘ABCDEFGHIGKLMNOP’) – 소문자로

SELECT **UPPER**(‘ABCDEFGHIGKLMNOP’) – 대문자로

SELECT **REPLACE**(‘MSSQL’, ‘MS’, ‘MY’) – MS를 MY로 바꿔줌

SELECT **TRIM**(‘ ASDS ‘) – 공백을 없애줌

SELECT TRIM(**LEADING** ‘#’ FROM ‘###MS###’) – 앞에 #이 있으면 없애줌

SELECT TRIM(**TRAILING** ‘#’ FROM ‘###MS###’) – 뒤에 #이 있으면 없애줌

SELECT **FORMAT**(123123123.123123123,5) – 소수점 5째 까지 출력하고 정수 부분은 세개씩 끊어 출력

SELECT **FLOOR**(10.95), **CEIL**(), **ROUND**() – 각각 내림, 올림 반올림

SELECT **SQRT**(10.95), **POW(), EXP(), LOG()** – 각각 제곱근, 거듭제곱, e의 거듭제곱, 자연로그

SELECT **SIN(), COS(), TAN() -** 사인, 코사인, 탄젠트

SELECT **ABS**(), **RAND**() – 절대값, 랜덤값 **ROUND(RAND\*100,0)** – 1~100까지 랜덤(0은 소수자리)

SELECT **NOW(), CURDATE(), CURTIME() –** 현재 날짜 시간 반환, 현재날짜 반환, 현재시간 반환

SELECT **DATE**(NOW()), **MONTH**(NOW()), **DAY**(NOW()), **HOUR**(NOW()), **MINUTE**(NOW()), **SECOND**(NOW())

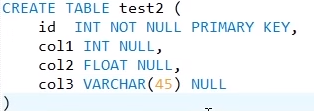
SELECT **MONTHNAME**(NOW()), **DAYNAME**(NOW()), **DATOFWEEK**(NOW()), **DAYOFYEAR**(NOW())

SELECT **DATE\_FORMAT**(NOW(), ‘%D %y %a %d %m %n %j’)

**SQL 고급**

**CREATE** **TABLE** CITY2 **AS** SELECT \* FROM CITY; - CITY와 똑같은 CITY2테이블 만듬

CREATE **DATABASE** LEE; - LEE라는 데이터 베이스 만듬



* 테이블 만들기 INT형식,NULL허용X,가장 중요한 키

**ALTER** TABLE TEST2 **ADD** COL4 INT NULL – 테이블에 열을 추가

**ALTER** TABLE TEST2 **MODIFY** COL4 FLOAT NULL – 테이블 열수정

**ALTER** TABLE TEST2 **DROP** COL4 – 열 삭제

SET @H := -1 - 변수만들기

**인덱스**

* 데이터 많으면 검색 오래걸려서 사용, 수정자주보다 검색자주 테이블에 사용이 유용

CREATE **INDEX** COL111 **ON** TEST(COL1); -TEST테이블의 COL1열에 COL111이라는 인덱스를 붙여줌(KEY네임)

**SHOW** INDEX FROM TEST – 테이블의 인덱스 보여줌

CREATE **UNIQUE** INDEX COL111 ON TEST(COL1); - 유니크한 열을 만들어줌

CREATE **FULLTEXT** INDEX COL111 ON TEST(COL1); - 풀텍스트는 일반인덱스보다 빠르게 탐색가능한 타입

ALTER TABLE TEST **DROP** INDEX COL2 – 인덱스 삭제

**DROP** INDEX COL2 ON TEST – 인덱스 삭제

**VIEW**

* 가상 테이블, 내가 원하는 값들만 보여줄 수 있음

**CREATE VIEW** TESTVIEW AS SELECT COL1,COL2 FROM TEST- 뷰 생성

**ALTER VIEW** TESTVIEW AS SELECT COL1,COL2 FROM TEST- 뷰 수정

**DROP** VIEW – 뷰 삭제

**INSERT** INTO TEST VALUE(1,123,13,) – 테이블에 값 삽입

**INSERT** INTO TEST SELECT \* FROM TEST2 – TEST2를 복사하여 TEST에 넣어줌

**UPDATE** TEST SET COL1 = 1, COL2 = 1 WHERE ID = 1 – 값 업데이트(WHERE안쓰면 다 수정됨)

**DELETE** FROM TEST WHERE ID=1 – 값 삭제, 복구가능

**TRUNCAET** TABLE- 삭제인데 되돌릴 수 없음

**DROP** TABLE – 삭제, 되돌릴 수 없음

**DROP** DATABASE – 삭제, 되돌릴 수 없음