

# DBMS

## 7일차

강사 : 김근형

## ◆ ORDER BY 구문

- SELECT 문을 사용할 때 출력되는 결과물은 테이블에 입력된 순서대로 출력되는 것이 기본
- 내림차순 혹은 오름차순으로 정렬된 데이터들이 필요할 때 ORDER BY 사용
- SELECT 문의 마지막에 위치
- 디폴트 값은 오름차순(ASC) / 내림차순(DESC)
- 구조

SELECT \* FROM 테이블명 ORDER BY 컬럼명 ASC or DESC;

## ◆ ORDER BY 구문의 사용 방법들

- 기본 형태

SELECT \* FROM 테이블명 ORDER BY 컬럼명;

- 컬럼 합산 사용

SELECT 컬럼1+컬럼2 AS TOTAL FROM 테이블명 ORDER BY TOTAL;

- 컬럼의 숫자를 사용(숫자 번째 컬럼)

SELECT \* FROM 테이블명 ORDER BY 3; → 3번째 컬럼을 기준으로 정렬

- 여러 컬럼을 사용

SELECT \* FROM 테이블명 ORDER BY 3,1 DESC;

→ 3번째 컬럼을 기준으로 오름차순 정렬한 상태에서 1번째 열을 기준으로 내림차순 정렬

## ◆ LIMIT 구문

- SELECT 구문의 결과 레코드 수를 지정하는 것

- 구조

SELECT \* FROM 테이블명 LIMIT 5; → 5개만 출력

## ◆ LIMIT 구문

- SELECT 구문의 결과 레코드 수를 지정하는 것

- 구조

SELECT \* FROM 테이블명 LIMIT 5; → 5개만 출력

SELECT \* FROM 테이블명 LIMIT 2. 3 → 3번째 결과부터 3개만 출력(0부터 시작하므로 2)

## ◆ DISTINCT 구문

- 결과 값들 중 중복이 있을 시 제거하는 구문

- 구문

SELECT DISTINCT \* FROM 테이블명;

Tip)

SELECT COUNT(\*) FROM 테이블명; → 결과 값의 개수 확인할 수 있음

## ◆ GROUP BY 구문

- 특정 컬럼을 기준으로 집계 함수를 사용하여 건수(COUNT), 합계(SUM), 평균(AVG) 등 집계성 데이터를 추출할 때 사용
- 기준 컬럼을 여러 개 지정할 수 있으며, HAVING 구문을 함께 사용하여 WHERE 구문처럼 조건을 부여할 수 있다
- 중복 제거를 할 때도 사용한다
- 자동으로 ORDER BY가 되지만 명시적으로 ORDER BY를 입력하여 작성하는 것이 좋다
- 구문

SELECT 집계함수(컬럼) FROM 테이블명 WHERE 조건 GROUP BY 컬럼;

SELECT 집계함수(컬럼) FROM 테이블명 WHERE 조건 GROUP BY 컬럼 HAVING 조건;

## ◆ 서브 쿼리

- 하나의 SQL 문에 포함되어 있는 또 다른 SQL 문
- 괄호로 감싸서 사용
- 비교 연산자와 함께 사용 가능
- 분류
  - 단일 행 서브 쿼리
  - 다중 행 서브 쿼리



## ◆ 단일 행 서브 쿼리

- 서브 쿼리의 수행 결과가 오직 하나의 결과만을 반환
- 비교를 통해 쿼리 수행
- 사용하는 비교 연산자
  - $>$ ,  $>=$ ,  $<$ ,  $<=$ ,  $=$ , ....

## ◆ 다중 행 서브 쿼리

- 서브 쿼리의 결과가 두 건 이상의 결과를 반환
- 다중 행 비교연산자 사용
  - IN, ANY, SOME, ALL, EXISTS 등..

## ◆ 다중 조건문

- if 조건을 여러 개 사용하는 구문

- 구문

```
SELECT IF(조건, 참, IF(조건, 참, 거짓)) FROM 테이블명;
```