



---

# SQL

데이터베이스

---

---

# 데이터베이스

- 특정 조직의 기능을 수행하는데 필요한 데이터의 모음
  - 여러 사람이 공동으로 사용할 수 있도록 데이터를 중복을 최소화 하여 보관
  - 특징
    - 실시간 접근성
    - 계속적인 변화
    - 동시 공유
    - 내용에 의한 참조
    - 중복 최소화
-

---

# DBMS

- Data Base Management System 데이터관리시스템
- 사용자와 데이터베이스 사이에서 위치하여 데이터베이스를 관리함
- 필수기능
  - 정의 기능
  - 조작 기능
  - 제어 기능

---

# 스키마

- 데이터베이스를 구성하는 개체, 속성, 관계 등 구조에 대한 정의와 이에 대한 제약 조건 등을 기술
- 종류
  - 외부 스키마 : 일반 사용자나 응용 프로그램 관점
  - 개념 스키마 : 기관 조직체 관점
  - 내부 스키마 : 시스템 프로그래머, 설계자 관점

---

# 데이터베이스 사용자

- 데이터베이스 관리자(DBA) : 데이터베이스 시스템을 관리하고 운영 책임자, 데이터베이스 스키마 생성/정의/삭제, 데이터 사전 유지관리
  - 응용 프로그래머 : 일반 프로그래밍언어(자바, 파이썬, 자바스크립트)로 작성된 프로그램에 데이터 조작용어 넣어 데이터베이스를 접근
  - 일반 사용자 : 질의어를 사용하여 데이터베이스 접근
-

---

# 데이터베이스 종류

- 계층형 데이터베이스
  - 망 데이터베이스
  - 관계형 데이터베이스
  - 객체 지향형 데이터베이스
  - NoSQL
-

# 관계형 데이터베이스 구성요소

속성명					튜플
개체 인스턴스	학번	이름	학년	학과	
	201101	홍길동	1학년	컴퓨터공학과	
	201102	이길동	2학년	미술과	
	201103	삼길동	3학년	기계공학과	
	201104	오길동	4학년	수학과	
속성					도메인

---

# 관계형 데이터베이스 구성요소

- 테이블
    - 데이터를 행과 열로 표현, 튜플(레코드)의 집합
    - 릴레이션 스키마는 일정 수의 속성 집합
  - 튜플 : 테이블 행을 구성하는 개체
  - 속성 : 테이블의 열을 구성하는 항목, 데이터베이스를 구성하는 가장 작은 단위
  - 도메인: 속성 하나에 취할 수 있는 값의 범위
  - 차수 : 속성의 개수
  - 기수 : 튜플의 개수
-



---

# 키(Key)의 개념 종류

- 키는 데이터베이스 조건에 만족하는 레코드를 찾거나 순서대로 정렬할 때 기준이 되는 속성

학번	이름	주민번호
201101	홍길동	1904201020310
201102	이길동	1904201020311
201103	삼길동	1904201020312
201104	오길동	1904201020313

학번	과목명
201101	수학
201102	영어
201103	과학
201104	프랑스어

---

---

# 키(Key)의 종류

- 후보키 : 테이블을 구성하는 튜플 중에서 유일하게 식별하기 위해 사용되는 속성
  - 기본키 : 한 릴레이션에서 특정 레코드를 유일하게 구별할 수 있는 속성
  - 외래키 : 관계를 맺고 있는 R1, R2 테이블에서 R1이 참조하고 있는 테이블 R의 기본 키와 같은 R1 테이블의 속성을 외래키라고 함
  - 대체키 : 후보키 중에서 기본키를 제외한 나머지 속성
-

---

# SQL의 개념

- 1974년 IBM 연구소에서 개발한 SEQUEL에서 유래
- 질의어, 질의 기능 뿐만 아니라 데이터 구조의 정의, 조작, 데이터 제어 기능을 제공

---

# SQL 종류

- DDL(데이터 정의어)
    - CREATE, ALTER, DROP
  - DML(데이터 조작어)
    - SELECT, INSERT, DELETE, UPDATE
  - DCL(데이터 제어어)
    - COMMIT, ROLLBACK, GRANT, REVOKE
-

---

# SQL

**SELECT** [DISTINCT] 필드이름 **FROM** 테이블이름 [**WHERE** 조건식];

- DISTINCT : 검색 결과 중복되는 레코드를 한번만 표현한다.
  - 필드이름 : 테이블의 모든 필드를 검색할 경우 \* 입력, 특정 필드만 입력할 경우 콤마(,)로 구분
  - WHERE 조건식 : 조건을 입력하여 특정 레코드만 검색할 때 사용
-

---

# 정렬

**SELECT** [DISTINCT] 필드이름 **FROM**  
테이블이름 [**WHERE** 조건식]  
[**ORDER BY** 필드이름 정렬방식...];

- ORDER BY 구문 : 특정 필드를 기준으로 레코드 정렬
  - 정렬 방식 : ASC와 DESC가 있음
-

---

# 그룹 지정

**SELECT** [DISTINCT] 필드이름 **FROM**  
테이블이름 [**WHERE** 조건식]  
[**GROUP BY** 필드이름]  
[**HAVING** 그룹조건식];

- GROUP BY 절 : 특정 필드를 기준으로 그룹화하여 검색
  - HAVING 절 : 그룹에 대한 조건을 지정할 때 사용
  - GROUP BY는 SUM, AVG, COUNT와 같은 그룹함수와 사용
-

---

# INSERT

**INSERT INTO** 테이블이름(속성명1, 속성명2...)  
**VALUES**(값1, 값2 ...);

- 대응하는 속성과 데이터는 개수와 데이터 형식이 일치
  - 기본 테이블의 모든 속성을 사용할 때에는 속성명 생략 가능
  - SELECT 문을 이용하여 다른 테이블의 검색 결과를 삽입할 수 있음
-



---

# DELETE

**DELETE FROM 테이블이름**  
**[WHERE 조건];**

- 모든 튜플을 삭제할 때에는 WHERE 조건절 생략
- 모든 튜플을 삭제하더라도 테이블 구조는 남아있음

---

# UPDATE

**UPDATE** 테이블이름  
**SET** 속성명1 = 값1, 속성명2 = 값2 ...  
**[WHERE 조건];**

- 기본 테이블에 있는 튜플 중에서 특정 튜플의 내용을 변경시킬 때 사용

---

# CREATE TABLE

```
CREATE TABLE 테이블이름(  
    속성이름1 자료형 [옵션],  
    속성이름2 자료형 [옵션],  
    속성이름3 자료형 [옵션],  
    속성이름4 자료형 [옵션],  
    ...  
);
```

- 1. DEFAULT 기본값
- 2. NOT NULL
- 3. PRIMARY KEY
- 4. UNIQUE
- 5. FOREIGN KEY(외래키\_속성)  
REFERENCES 참조테이블(기본키\_속성)

---

# CREATE VIEW

**CREATE VIEW** 뷰이름(속성명1, 속성명2, ...)  
**AS SELECT 문;**

- SELECT 문으로 쿼리를 하고 그 결과로 뷰를 생성
  - 테이블이 물리적, 뷰는 논리적
-

---

# CREATE INDEX

**CREATE [UNIQUE] INDEX 인덱스이름  
ON 테이블명(속성이름1 [ASC|DESC], 속성이름2) ;**

- UNIQUE : 중복 값이 없는 고유한 특성을 갖는 인덱스 생성
  - ASC | DESC : 오름차순, 내림차순 정렬 지원
-

---

# ALTER TABLE

**ALTER TABLE 테이블명 ADD COLUMN 속성명 데이터\_타입;**  
**ALTER TABLE 테이블명 RENAME COLUMN 속성명1 TO 속성명2;**  
**ALTER TABLE 테이블명 DROP COLUMN 속성명;**

- ADD : 새로운 속성(열)을 추가
  - RENAME : 속성 이름을 변경
  - DROP : 속성을 삭제
-

---

# DROP TABLE

**DROP TABLE** 테이블명;

- 테이블 삭제