

RDBMS

관계형 데이터베이스 시스템 (RDBMS)

강사 양석환



RDBMS 개요

AiDA
Lab.

• 기본 용어

• 데이터

- 하나하나의 단편적인 정보. 정보는 있으나 아직 체계화 되지 못한 상태

• 테이블

- 데이터를 입력하기 위해, 표 형태로 표현한 것. (Ex) 회원 정보 테이블, 제품 정보 테이블

• 데이터베이스(DB)

- 테이블이 저장되는 저장소(각 데이터베이스는 서로 다른 고유한 이름을 가지고 있음)

- **DBMS (DataBase Management System)**
 - 데이터베이스를 관리하는 시스템 또는 소프트웨어
- **SQL (Structured Query Language)**
 - 구조화된 질의 언어. 사람과 DBMS가 소통하기 위한 말(언어)



- **데이터베이스의 정의와 특징**

- **데이터베이스**

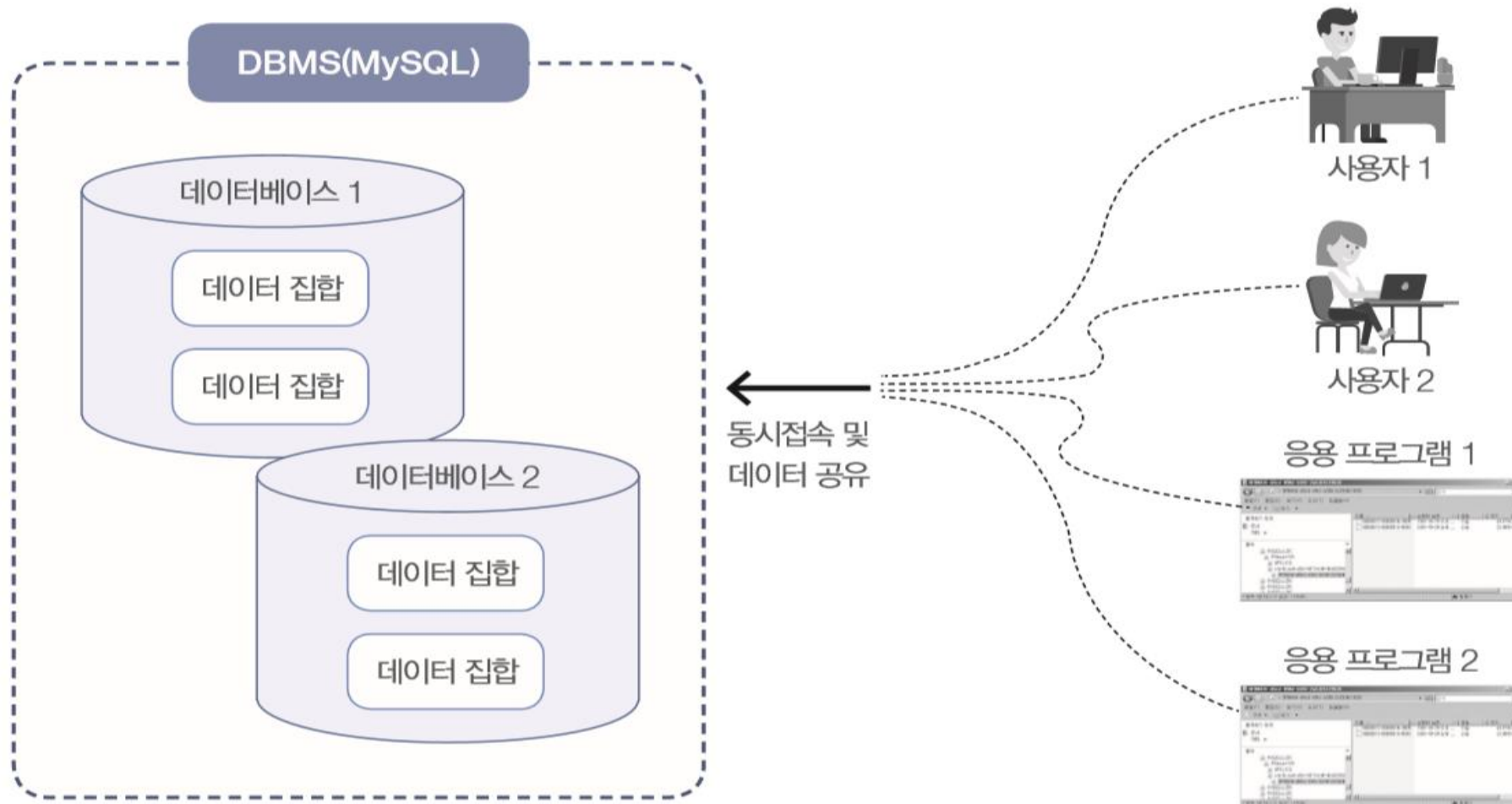
- 데이터의 집합
 - 여러 명의 사용자나 응용프로그램이 공유하는 데이터들
 - 동시에 접근 가능해야
 - 데이터의 저장 공간 자체

- **DBMS**

- 데이터베이스를 관리·운영하는 역할



• DBMS 개념도



- **DB/DBMS의 특징**

- **데이터의 무결성 (Integrity)**

- 데이터베이스 안의 데이터는 오류가 없어야 함
 - 제약 조건(Constrain)이라는 특성을 가짐

- **데이터의 독립성**

- 데이터베이스 크기 변경하거나 데이터 파일의 저장소 변경 시 기존에 작성된 응용프로그램은 전혀 영향을 받지 않아야 함

- **보안**

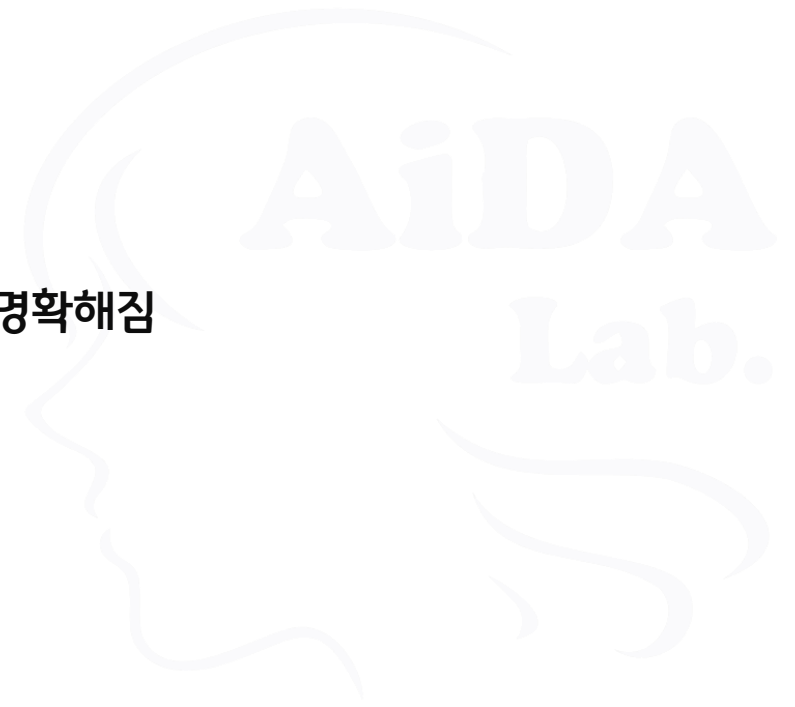
- 데이터베이스 안의 데이터에 데이터를 소유한 사람이나 데이터에 접근이 허가된 사람만 접근할 수 있어야 함
- 접근할 때도 사용자의 계정에 따라서 다른 권한 가짐

- **데이터 중복의 최소화**

- 동일한 데이터가 여러 개 중복되어 저장되는 것 방지



- 응용프로그램 제작 및 수정이 쉬워짐
 - 통일된 방식으로 응용프로그램 작성 가능
 - 유지보수 또한 쉬워짐
- 데이터의 안전성 향상
 - 대부분의 DBMS가 제공하는 백업·복원 기능 이용
 - 데이터가 깨지는 문제가 발생할 경우 원상으로 복원 , 복구하는 방법이 명확해짐



- 초기

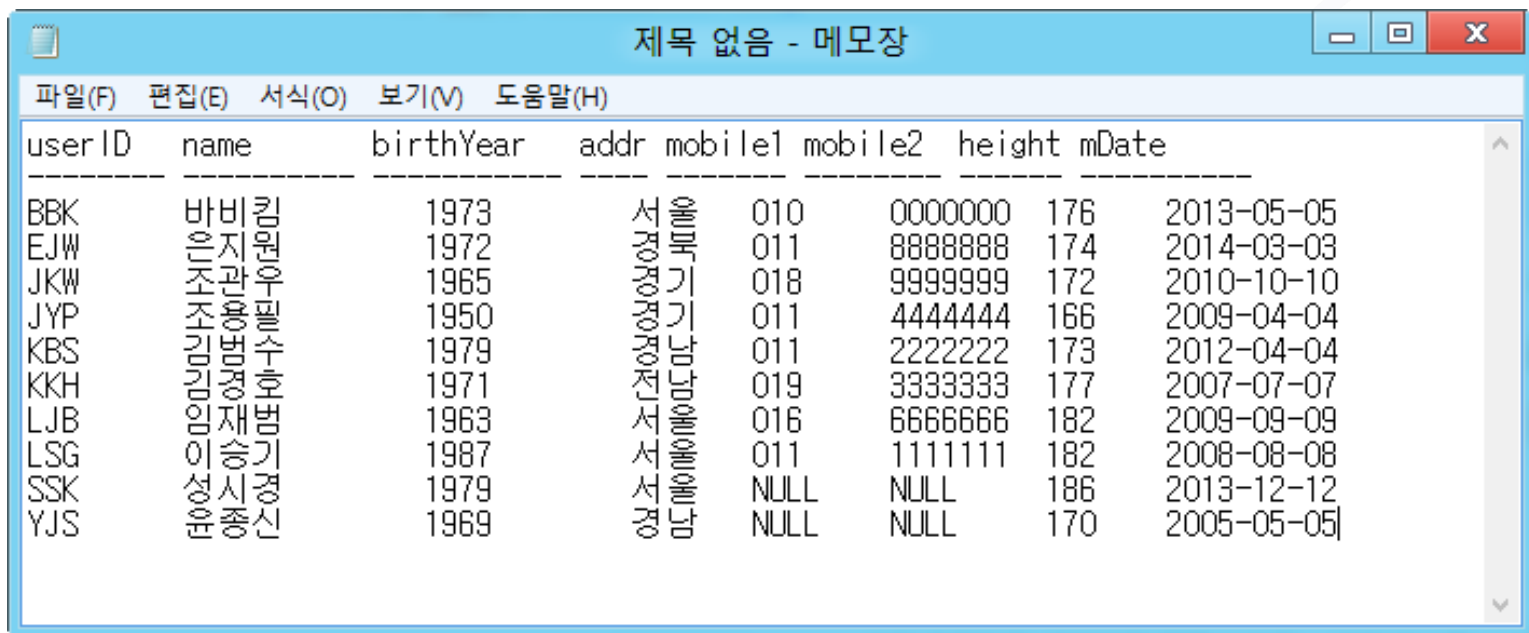
- 오프라인 관리
- 종이에 연필로 기록해 장부로 관리



[그림 1-2] 종이 장부

• 파일시스템 사용

- 컴퓨터 파일에 기록/저장 - 메모장, 엑셀 활용
- 컴퓨터에 저장된 파일의 내용은 읽고, 쓰기가 편한 약속된 형태의 구조 사용
- 데이터의 양이 많아지면 데이터 중복으로 인한 불일치 위험



The screenshot shows a Notepad window with a menu bar (File, Edit, Format, View, Help) and a table of user data. The table has columns: userID, name, birthYear, addr, mobile1, mobile2, height, and mDate. The data is as follows:

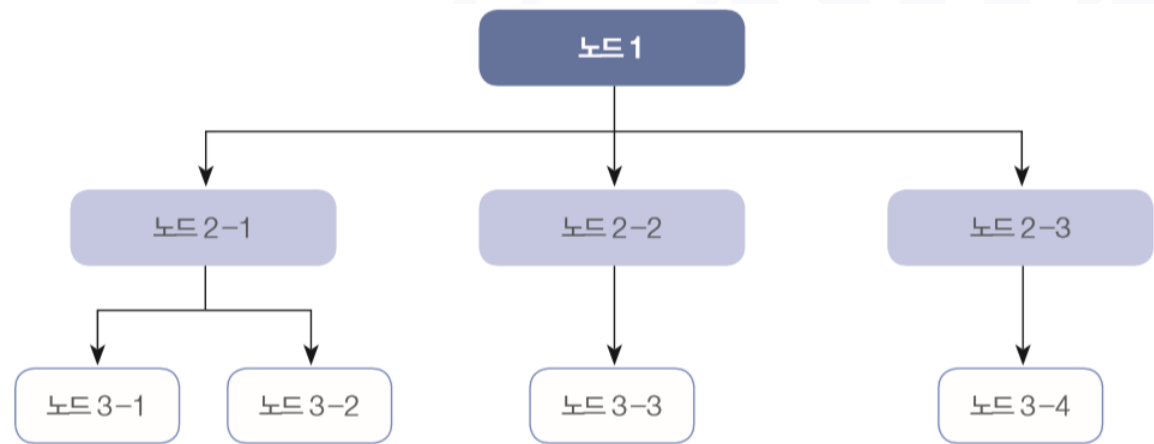
userID	name	birthYear	addr	mobile1	mobile2	height	mDate
BBK	바비킴	1973	서울	010	0000000	176	2013-05-05
EJW	은지원	1972	경북	011	8888888	174	2014-03-03
JKW	조관우	1965	경기	018	9999999	172	2010-10-10
JYP	조용필	1950	경기	011	4444444	166	2009-04-04
KBS	김범수	1979	경남	011	2222222	173	2012-04-04
KKH	김경호	1971	전남	019	3333333	177	2007-07-07
LJB	임재범	1963	서울	016	6666666	182	2009-09-09
LSG	이승기	1987	서울	011	1111111	182	2008-08-08
SSK	성시경	1979	서울	NULL	NULL	186	2013-12-12
YJS	윤종신	1969	경남	NULL	NULL	170	2005-05-05

- 데이터베이스 관리시스템(DataBase Management System, DBMS)
 - 파일시스템의 단점 보완
 - 대량의 데이터를 보다 효율적으로 관리하고 운영하기 위해 사용
 - 데이터의 집합인 데이터베이스를 잘 관리하고 운영하기 위한 시스템 또는 SW
- SQL(Structured Query Language)
 - DBMS에 데이터 구축/관리/활용 위해서 사용되는 언어
 - DBMS를 통해 중요한 정보들을 입력, 관리, 추출



• 계층형 DBMS

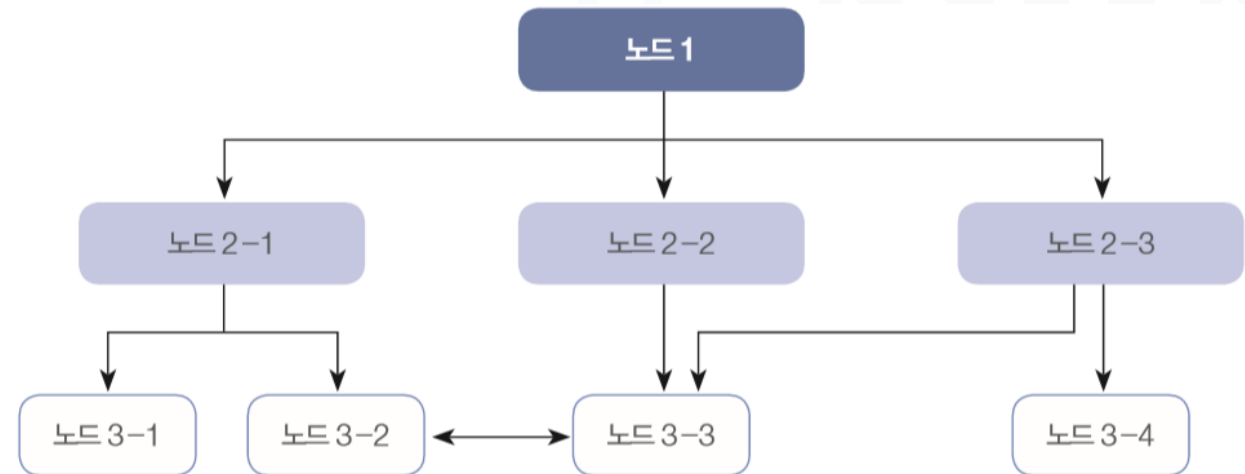
- 처음으로 나온 DBMS 개념 - 1960년대에 시작
- 각 계층은 트리Tree 형태, 1:N 관계
- 문제점
 - 처음 구축한 이후 그 구조를 변경하기가 상당히 까다로움
 - 주어진 상태에서의 검색은 상당히 빠름
 - 접근 유연성 부족해서 임의의 검색에는 어려움



[그림 1-4] 계층형 구조

• 망형 DBMS

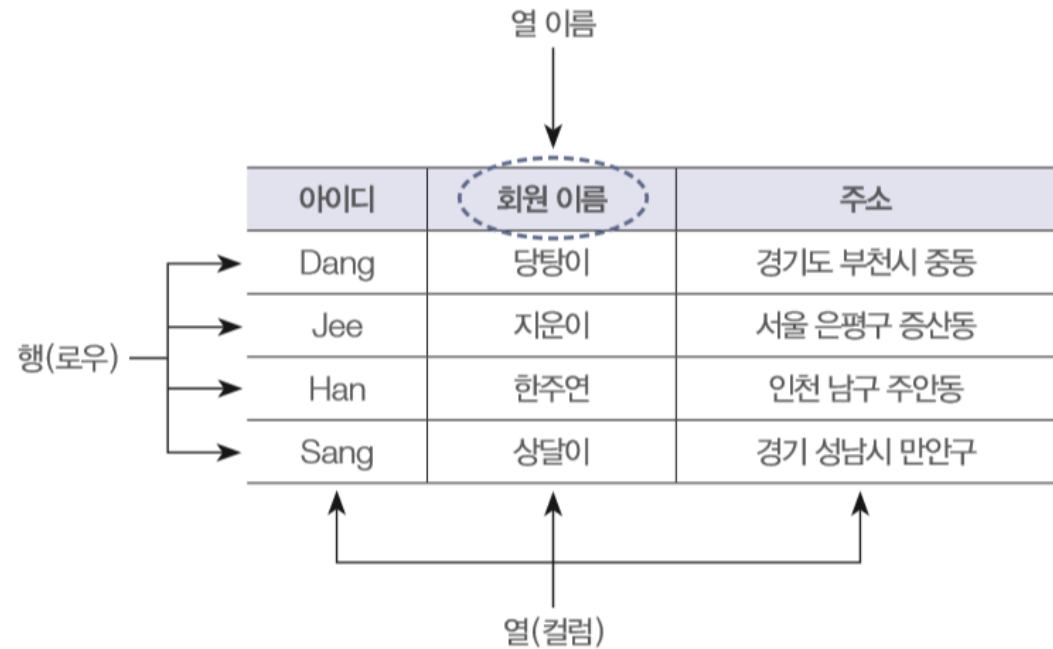
- 계층형 DBMS의 문제점을 개선하기 위해 1970년대에 시작
- 1:1, 1:N, N:M(다대다) 관계 지원 - 효과적이고 빠른 데이터 추출
- 복잡한 내부 포인터 사용
 - 프로그래머가 이 모든 구조를 이해해야만 프로그램의 작성 가능



[그림 1-5] 망형 구조

• 관계형 DBMS (Relational DBMS)

- 1969년 E.F.Codd라는 학자가 수학 모델에 근거해 고안
- 데이터베이스는 테이블Table이라 불리는 최소 단위로 구성
- 이 테이블은 하나 이상의 열로 구성



[그림 1-6] 관계형 구조

• 장점

- 다른 DBMS에 비해 업무가 변화될 경우 쉽게 변화에 순응
- 유지보수 측면에서도 편리
- 대용량 데이터의 관리와 데이터 무결성Integration보장

• 단점

- 시스템 자원을 많이 차지해 시스템이 전반적으로 느려지는 것
 - 하드웨어가 발전하면서 해결됨



RDBMS의 운영

AiDA
Lab.

- 분석, 설계, 구현, 시험, 유지보수의 5가지 단계

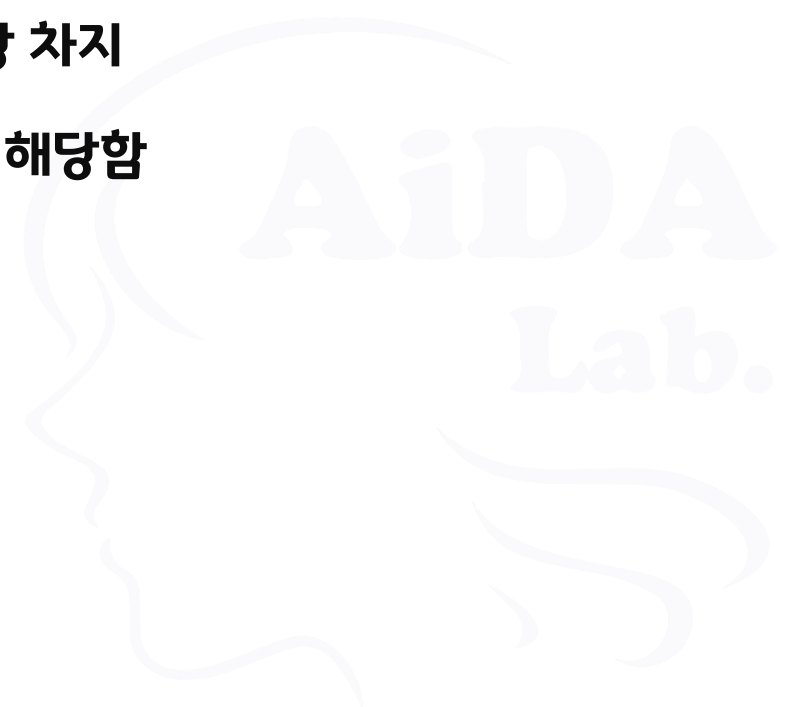
- 분석

- 구현하고자 하는 프로젝트의 가장 첫 번째 단계
- 시스템 분석 또는 요구사항 분석이라고 불림
- 요구사항 분석은 현재 우리가 '무엇을(What)' 할 것인지 결정
- 사용자의 인터뷰와 업무 조사 등을 수행
- 프로젝트의 첫 단추를 끼우는 중요한 단계
- 분석의 결과로 많은 문서 작성



• 설계

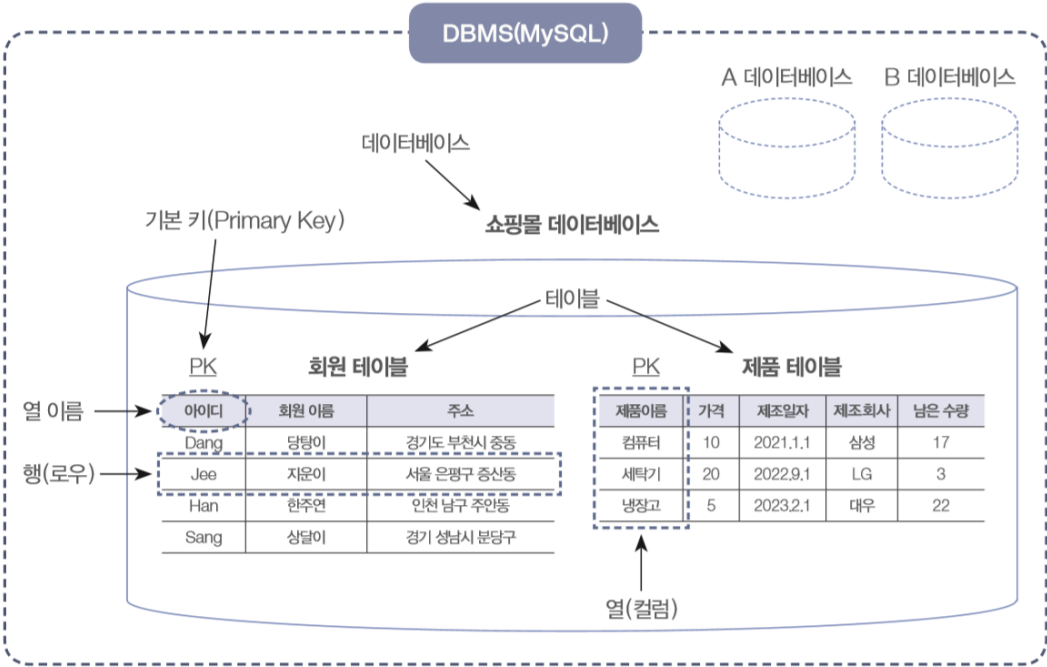
- 시스템 설계 또는 프로그램 설계
- 구축하고자 하는 시스템을 '어떻게(How)' 할 것인지 결정
- 대부분의 프로젝트에서 분석과 설계의 과정이 전체 공정의 50% 이상 차지
- 데이터베이스의 모델링, 구축, 운영은 정보시스템 구축단계의 일부에 해당함



- 구현

- 데이터베이스 모델링

- 현실세계에서 사용되는 데이터를 MySQL에 어떻게 옮겨 놓을 것인지를 결정하는 과정
- 저장할 정보는 테이블(Table)이라는 형식에 맞춰 저장
- Ex) 쇼핑몰 데이터 베이스의 예



[그림 3-1] 테이블의 구조와 관련 용어

- 데이터베이스 모델링 필수 용어

- 열(=컬럼=필드)

- 각 테이블은 열로 구성 (Ex. 회원 테이블의 경우에는 아이디, 회원 이름, 주소 등 3개의 열로 구성)

- 열 이름

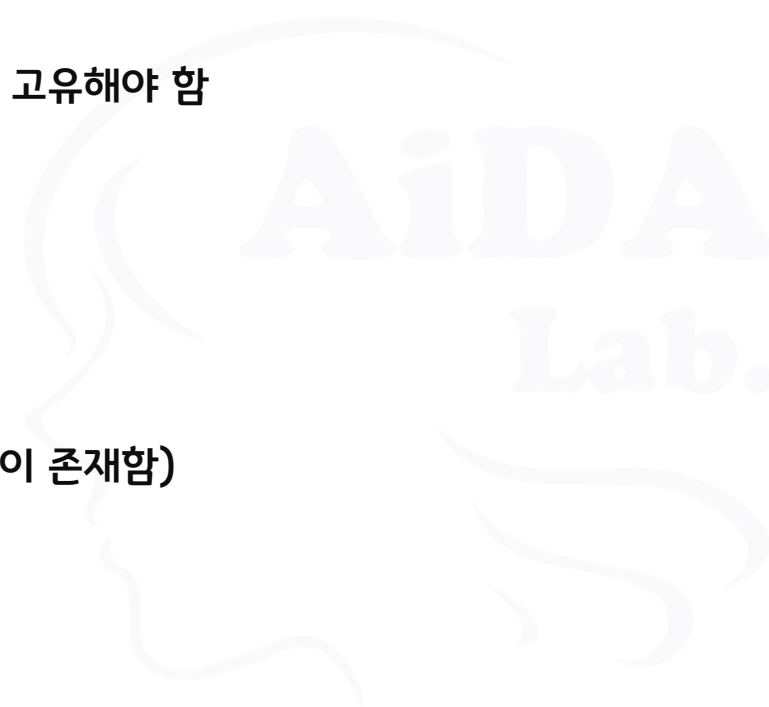
- 각 열을 구분하기 위한 이름. 열 이름은 각 테이블 내에서는 중복되지 않고, 고유해야 함

- 데이터 형식

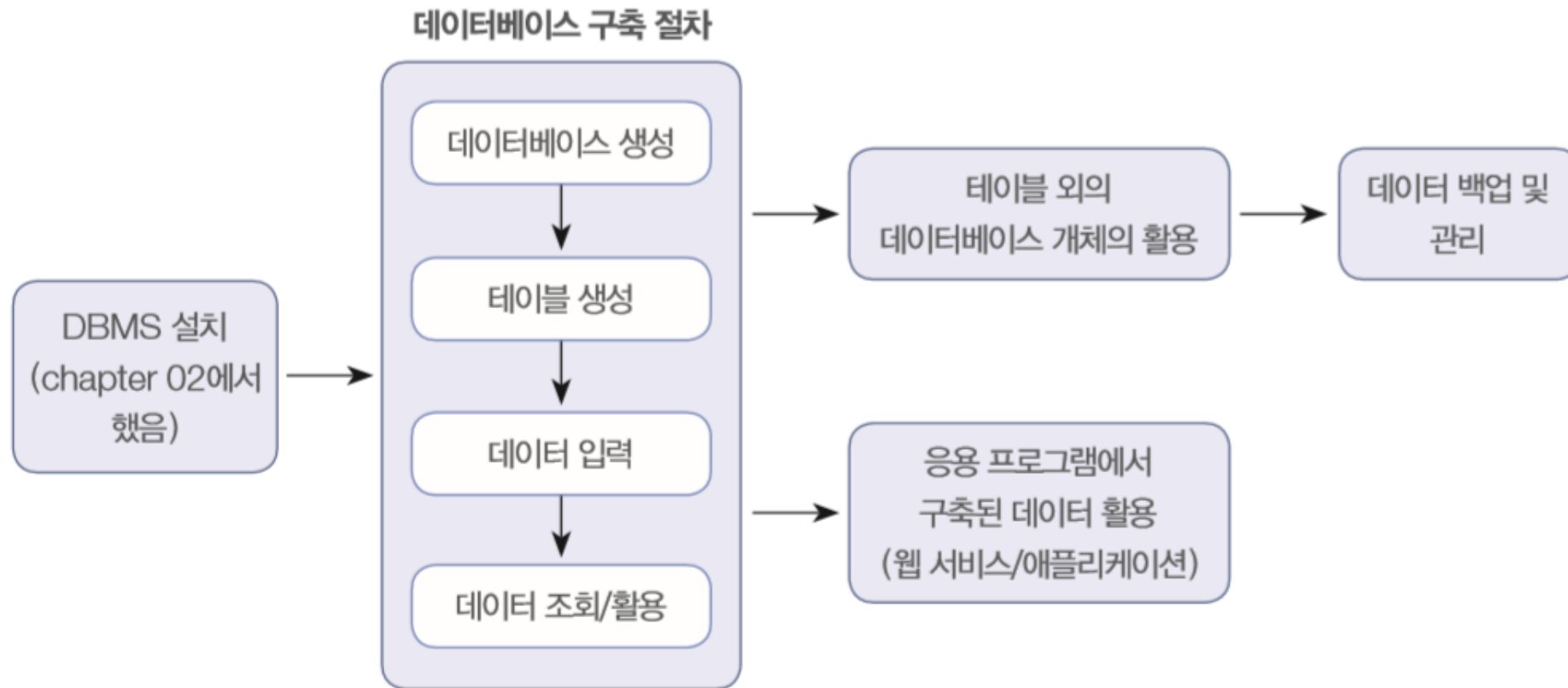
- 열의 데이터 형식. 테이블을 생성할 때 열 이름과 함께 지정

- 행(=로우=레코드)

- 실질적인 데이터 (Ex. 회원 테이블의 경우 4건의 행 데이터, 즉 4명의 회원이 존재함)



- 데이터베이스 구축/관리 및 활용의 전반적인 절차

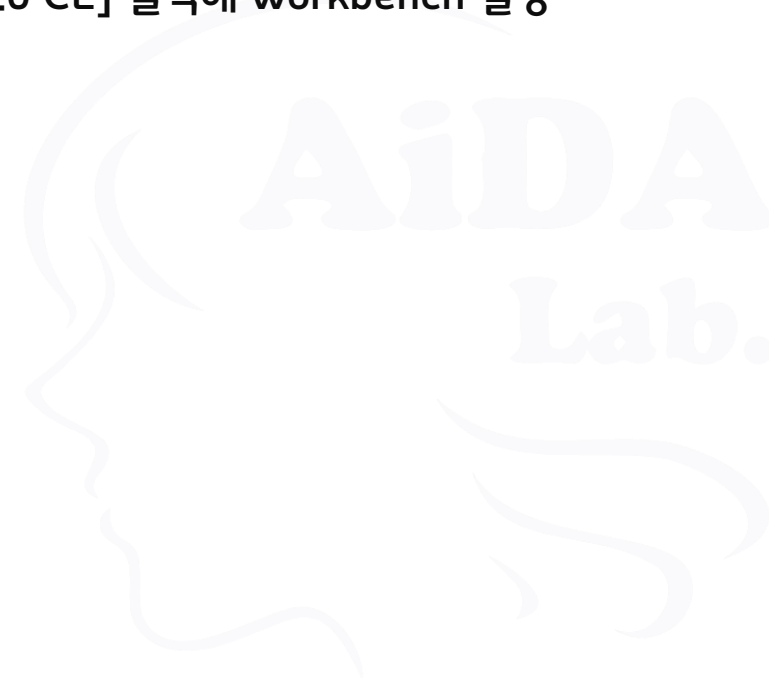


- 인터넷 쇼핑몰 구축 위한 ‘쇼핑몰’ DB 구축 예시

- 데이터베이스 생성

- MySQL 서버 연결 및 설정

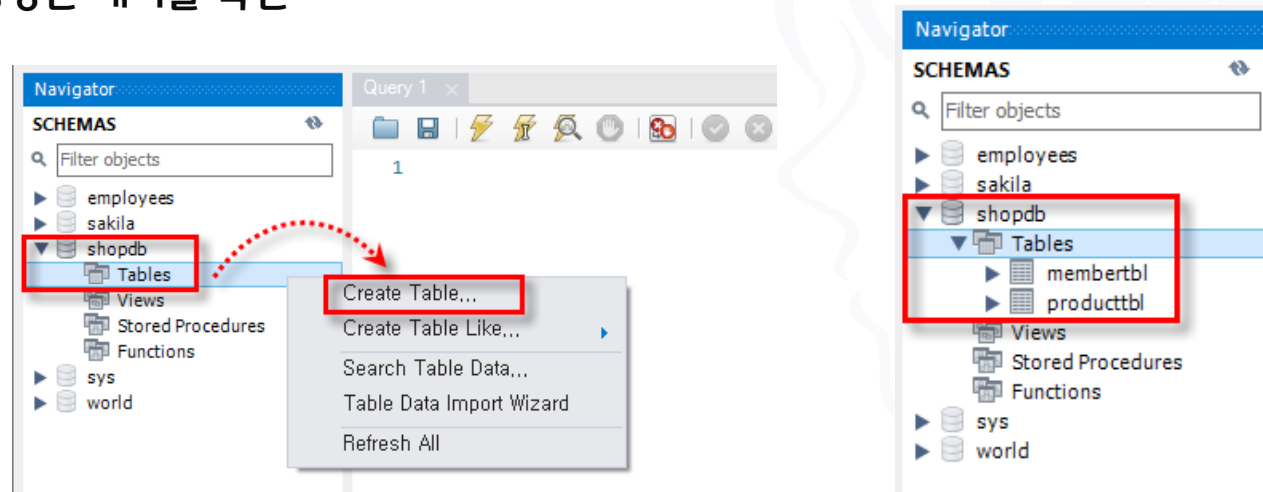
- Windows의 [시작] >> [M] >> [MySQL] >> [MySQL Workbench 8.0 CE] 클릭해 Workbench 실행
- [MySQL Connections] 창에서 비밀번호 입력하여 접속
- Workbench의 초기 창
 - 기본적으로는 [Schemas] 탭 클릭해 놓고 사용
- Workbench 종료 → 설정 저장



- 스키마 (Schema) 생성
 - MySQL에서는 스키마와 데이터베이스가 완전히 동일한 용어로 사용
 - Workbench의 [SCHEMAS]의 빈 부분
 - 마우스 오른쪽 버튼 클릭 후 [Create Schema](=Create Database) 선택
 - CREATE SCHEMA 'shopdb'문을 쿼리 창에서 입력하는 것과 동일한 작동
 - 이름 입력하면 DB 생성
 - 왼쪽 데이터베이스 목록에 shopdb 데이터베이스 확인
 - 아무것도 들어있지 않은 데이터베이스 생성

shopdb 는 생성하고자 하는 데이터베이스의 이름의 예시

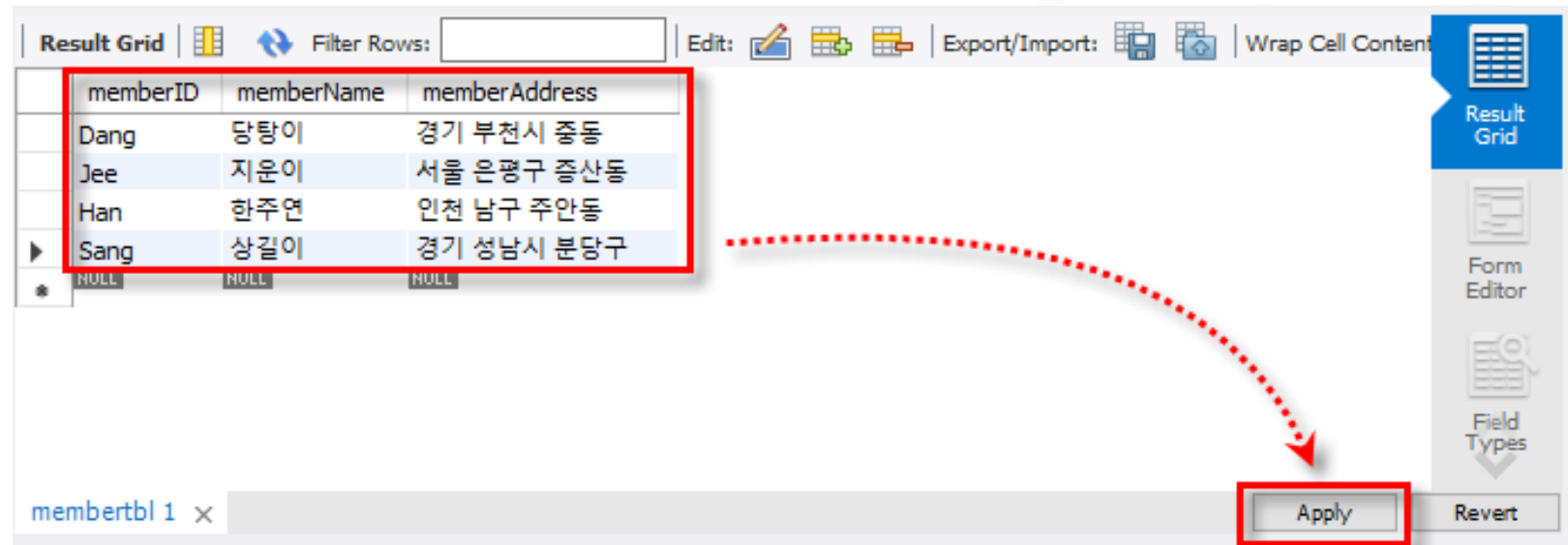
- 테이블 생성
 - 회원테이블, 제품 테이블 각 열의 영문 이름 및 데이터 형식 결정
 - 데이터베이스 모델링(특히 물리적 모델링) 시에 결정
 - 데이터 형식의 자세한 내용은 7장에서 학습
 - 형식이 정해지면 Create Table 실행해 테이블이름, 열이름, 데이터형식등 테이블 내용 입력
 - 생성된 SQL Query를 데이터베이스에 적용해 테이블 생성 완료
 - ShopDB의 [테이블]에서 생성한 테이블 확인



- 데이터 입력 – 행 데이터 입력

- 회원 테이블의 데이터 입력

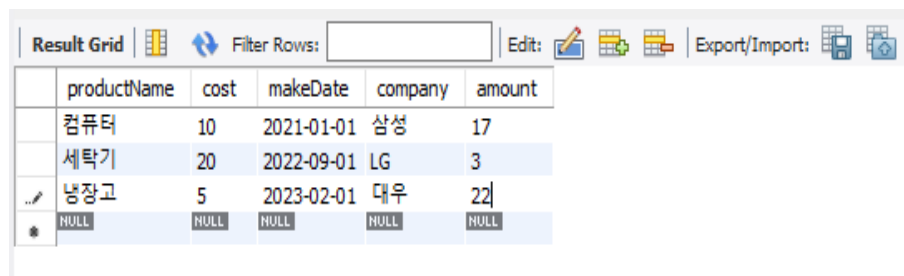
- Navigator의 [SCHEMAS]에서 [Shopdb] >> [Tables] >> [membertbl] 선택 후, 마우스 오른쪽 버튼 클릭하고 [Select Rows - Limits 1000] 선택
 - 아래 그림의 회원 테이블 데이터 Grid에 입력 후 Apply 해 저장



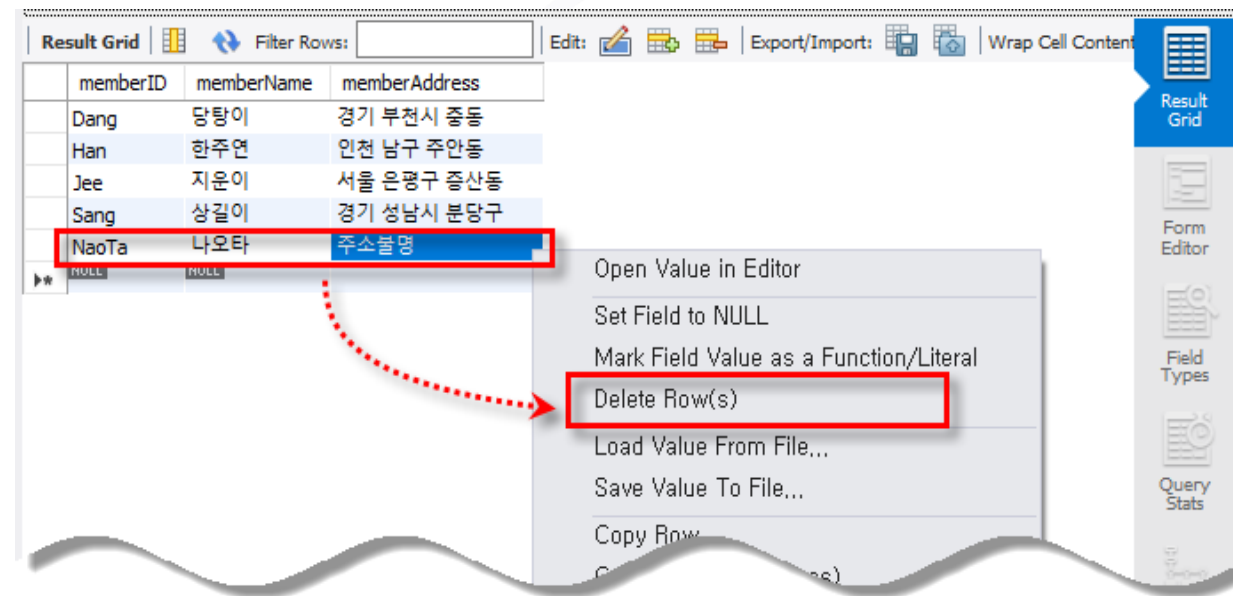
- 데이터 입력 – 행 데이터 입력

- 제품 테이블의 데이터 입력

- 동일한 방식으로 제품 데이터 입력 후 저장
 - 데이터를 삭제하려면 삭제할 행의 앞 부분에 마우스 대고 오른쪽 메뉴 Delete 사용해 삭제



	productName	cost	makeDate	company	amount
	컴퓨터	10	2021-01-01	삼성	17
	세탁기	20	2022-09-01	LG	3
	냉장고	5	2023-02-01	대우	22
	NULL	NULL	NULL	NULL	NULL



	memberID	memberName	memberAddress
	Dang	당탕이	경기 부천시 중동
	Han	한주연	인천 남구 주안동
	Jee	지운이	서울 은평구 증산동
	Sang	상길이	경기 성남시 분당구
	NaoTa	나오타	주소불명
	NULL	NULL	NULL

- Open Value in Editor
- Set Field to NULL
- Mark Field Value as a Function/Literal
- Delete Row(s)**
- Load Value From File...
- Save Value To File...
- Copy Row...
- Copy Rows...

- 데이터 활용

- 주로 SELECT 문 사용해 데이터 활용

- 사용할 데이터 베이스 선택

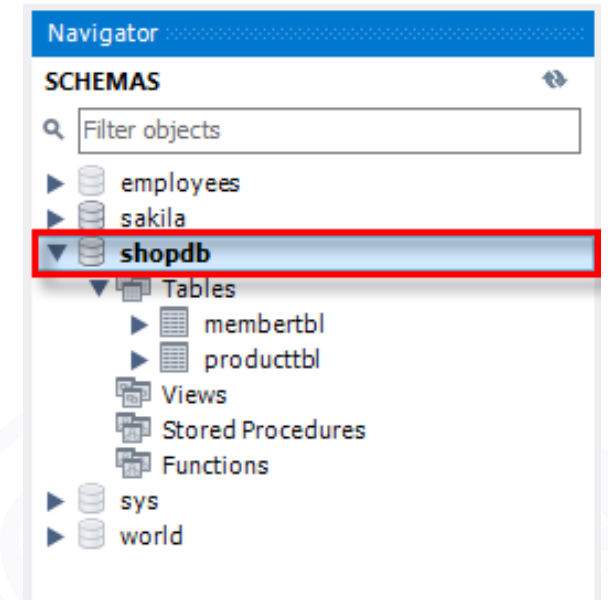
- SCHEMA에서 사용할 DB를 더블 클릭
 - 진하게 색상이 변하면서 선택 됨

- SQL 실행법

- 툴바의 <Execute the selected portion~~> 아이콘 클릭
 - [ctrl] + [shift] + [enter]
 - Workbench 메뉴의 [Query] >> [Execute(All or Selection)]

- SQL은 대소문자 구별 없음

- 읽기 편하게 예약어는 대문자 (쿼리 창에서 파란색으로 표시)



- SELECT 열 이름 FROM 테이블 이름 [WHERE 조건]
 - 모든 데이터 출력하기 (열 이름 대신 ' * ')
 - 열을 선택해 데이터 출력하기 (열 이름 나열)
 - 특정 데이터를 만족하는 데이터 출력하기 (WHERE절에 조건 입력)
- 테이블 삭제
 - DROP TABLE 테이블 이름



**THANK
YOU**

