

# 데이터베이스 기초 활용하기

## ❖ 자료와 정보

- ◆ 자료(Data) : 현실 세계에서 관찰이나 측정을 통해 수집한 단순한 사실이나 결과값으로, 가공되지 않은 상태
- ◆ 정보(Information) : 의사 결정에 도움을 줄 수 있는 유용한 형태로, 자료를 가공(처리)해서 얻는 결과물

❖ 자료 처리 시스템 : 정보 시스템이 사용할 자료를 처리하는 정보 시스템의 서브 시스템  
처리 형태에 따라

일괄 처리 시스템, 온라인 실시간 처리 시스템, 분산 처리 시스템  
으로 분류



❖ 데이터웨어 하우스(DataWare House) :

조직이나 기업체의 중심이 되는 주요 업무 시스템에서 추출되어  
새로이 생성된 데이터베이스로서 의사 결정 지원 시스템을 지원하는  
주체적, 통합적, 시간적 데이터의 집합체

## ❖ 데이터 베이스(Database/DB)

### ❖ 정 의 :

데이터베이스(영어: database, DB)는 체계화된 데이터의 모임이다.  
즉, 작성된 목록으로써 여러 응용 시스템들의 통합된 정보들을  
저장하여 운영할 수 있는 공용 데이터들의 묶음이다.

위키백과 : <https://ko.wikipedia.org/wiki/데이터베이스>

- ◆ 통합된 데이터(Integrated data) : **자료의 중복을 배제**한 데이터의 모임
- ◆ 저장된 데이터(Stored data) : 컴퓨터가 접근할 수 있는 **저장 매체에 저장된 자료**
- ◆ 운영 데이터(Operational data) : **조직의 업무를 수행하는 데 있어서**  
존재 가치가 확실하고 없어서는 안 될  
**반드시 필요한 자료**
- ◆ 공용 데이터(Shared Data) : 여러 응용 시스템들이 공동으로 소유하고  
유지하는 자료

## ❖ 데이터 베이스(Database/DB)

### ❖ 특 징 :

- ◆ 실시간 접근성(Real Time Accessibility) :  
수시적이고 비정형적인 질의(조회)에 대하여  
실시간 처리(Real-Time Processing)에 의한 응답이 가능함
- ◆ 계속적인 변화(Continuous Evolution) :  
새로운 데이터의 삽입(Insertion),  
삭제(Deletion),  
갱신(Update)으로  
항상 최신의 데이터를 유지함
- ◆ 동시 공유[동시 공유](Concurrent Sharing) :  
여러 사용자가 동시에 자기가 원하는 데이터를 이용할 수 있음
- ◆ 내용에 의한 참조(Content Reference) :  
데이터베이스에 있는 데이터를 참조할 때 데이터 주소나 위치에 의해서가 아니라  
사용자가 요구하는 데이터 내용으로 데이터를 찾음

## ❖ 정보 시스템

- ◆ 조직체에 필요한 **Data**를 수집, 저장해 두었다가 필요 시에 처리해서 의사 결정에 유용한 정보를 생성하고 분배하는 수단
- ◆ 사용하는 목적에 따라

경영 정보 시스템

군사 정보 시스템

인사 행정 정보 시스템

의사 결정 지원 시스템

등으로 사용된다.

## ❖ DBMS(DataBase Management System)

### ❖ 정 의 :

- ◆ **사용자와 데이터베이스 사이에서** 사용자의 요구에 따라 **정보를 생성**해주고,  
**데이터베이스를 관리**해주는 소프트웨어
- ◆ 기존의 파일 시스템이 갖는  
**데이터의 종속성과 중복성의 문제를 해결**하기 위해 제안된 시스템
- ◆ 데이터베이스의 구성, 접근 방법, 유지관리에 대한 모든 책임을 진다.

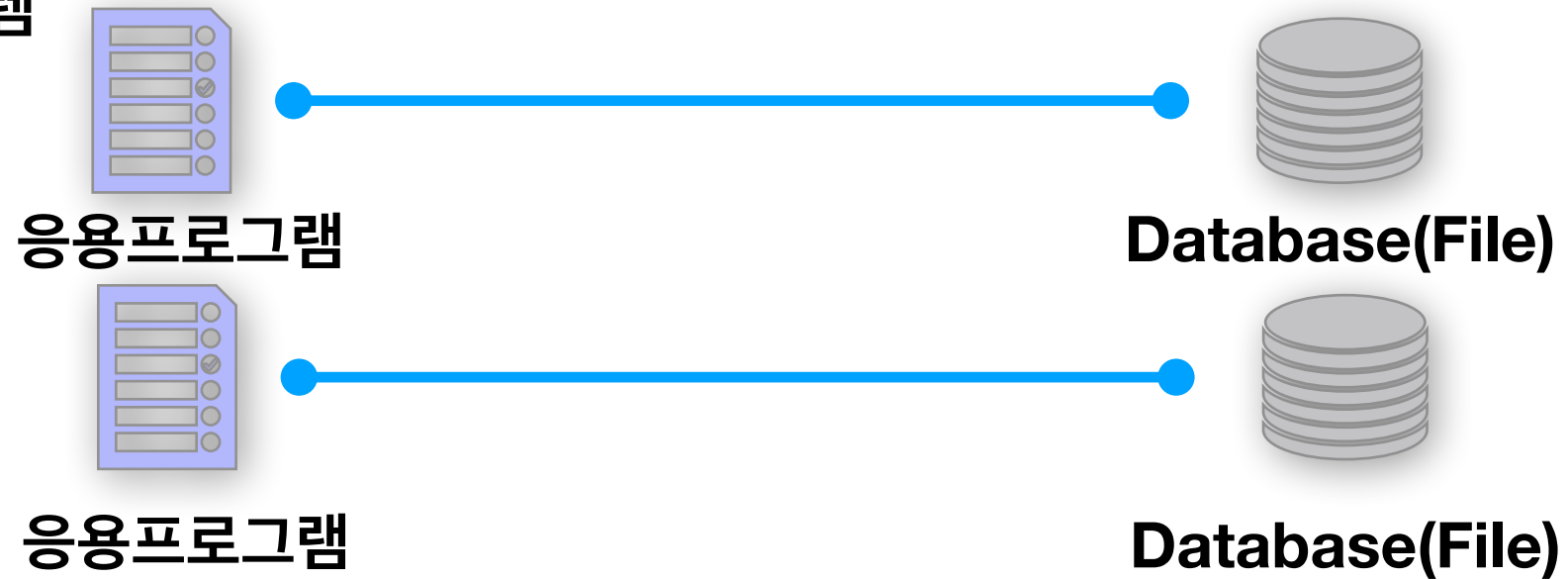
### \* 기존 파일 처리 방식의 문제점

- ◆ **종속성**
- ◆ **중복성**

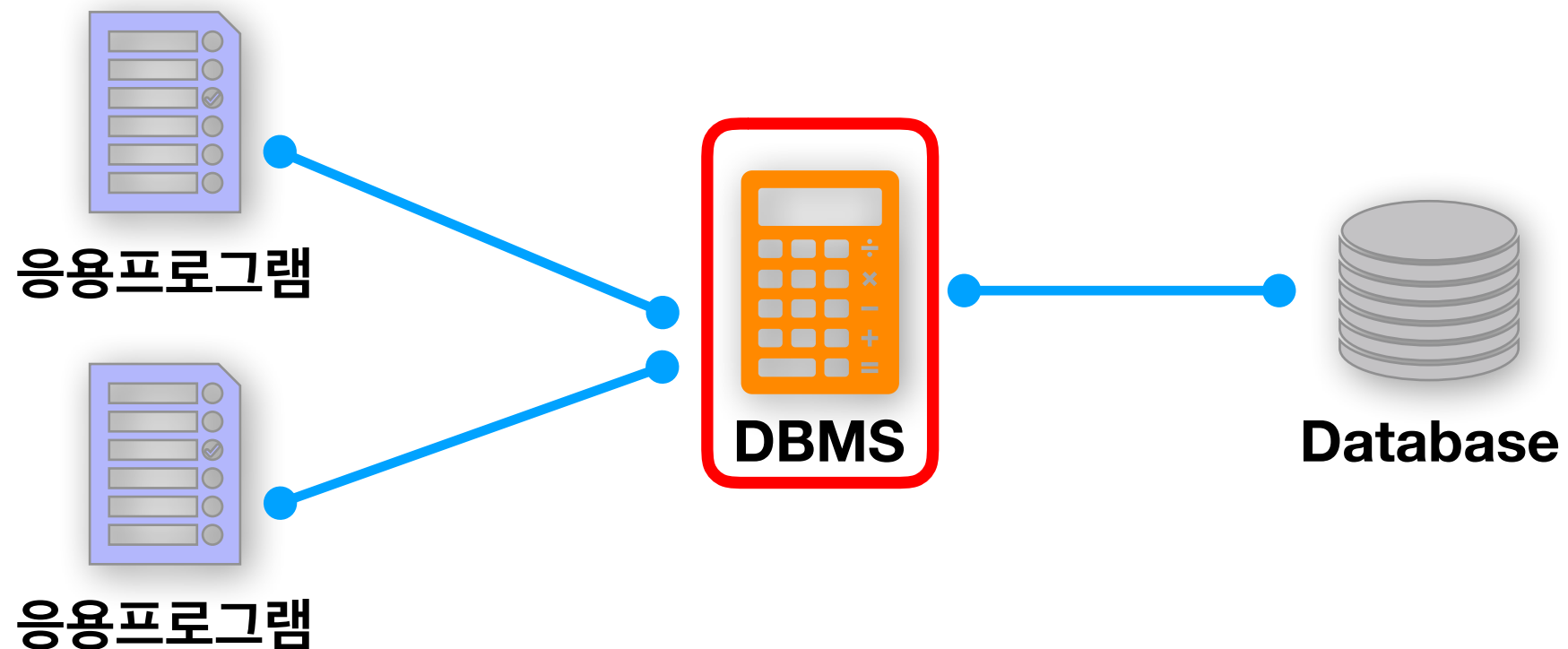
## ❖ DBMS(DataBase Management System)

### ❖ 파일 시스템과 DBMS의 비교

#### ◆ 파일 시스템



#### ◆ DBMS



## ❖ DBMS(DataBase Management System)

### ❖ 기존 파일 처리 방식의 문제점

#### ❖ 종속성으로 인한 문제점

- 종속성이란 응용 프로그램과 데이터 파일이 상호 의존적인 관계  
응용프로그램마다 데이터 파일이 존재
- 데이터 파일이 보조기억장치에 저장되는 방법이나  
저장된 데이터의 접근 방법을 변경할때는  
응용프로그램도 같이 변경해야 한다.

#### ❖ 중복성으로 인한 문제점

- 일관성 : 중복된 데이터간에 내용이 일치하지 않는 상황이 발생
- 보안성 : 중복되어 있는 모든 데이터에 동등한 보안 수준을 유지하기가 어려움
- 경제성 : 저장공간의 낭비와 동일한 데이터의 반복 작업으로 인한 비용 증가
- 무결성 : 제어의 분산으로 인해 데이터의 정확성을 유지할 수 없음



## ❖ DBMS(DataBase Management System)

### ❖ DBMS의 필수 기능 :

#### ◆ 정의(조직)(Definition)

- 데이터의 형(Type)과 구조,  
데이터가 DB에 저장될 때의 제약조건 등을 명시하는 기능
- 데이터와 데이터의 관계를 명확하게 명세할 수 있어야 하며,  
원하는 데이터 연산은 무엇이든 명세할 수 있어야 한다.

#### ◆ 조작(Manipulation)

- 데이터 검색(요청), 갱신(변경), 삽입, 삭제 등을 체계적으로 처리하기 위해  
데이터 접근 수단 등을 정하는 기능

#### ◆ 제어(Control)

- 데이터베이스를 접근하는 갱신 삽입, 삭제 작업이 정확하게 수행되어  
데이터의 무결성이 유지되도록 제어해야 한다.
- 정당한 사용자가 허가된 데이터만 접근할 수 있도록  
보안(Security)을 유지하고 권한(Authority)을 검사할 수 있어야 한다.
- 여러 사용자가 데이터베이스를 동시에 접근하여 데이터를 처리할 때  
처리 결과가 항상 정확성을 유지하도록  
병행 제어(Concurrency Control)를 할 수 있어야 한다.

## ❖ DBMS(DataBase Management System)

### ❖ DBMS의 장단점

장 점	단 점
<ul style="list-style-type: none"> <li>• 데이터의 중복을 피할 수 있음</li> <li>• 저장된 자료를 공동으로 이용할 수 있음</li> <li>• 데이터의 일관성을 유지할 수 있음</li> <li>• 데이터의 무결성을 유지할 수 있음</li> <li>• 보안을 유지할 수 있음</li> <li>• 데이터를 표준화할 수 있음</li> <li>• 데이터를 통합하여 관리할 수 있음</li> <li>• 항상 최신의 데이터를 유지함</li> <li>• 데이터의 실시간 처리가 가능함</li> <li>• 데이터의 논리적.물리적 독립성이 보장</li> </ul>	<ul style="list-style-type: none"> <li>• 데이터베이스 전문가 부족</li> <li>• 전산화 비용이 증가함</li> <li>• 대용량 디스크로의 집중적인 Access로 과부하(Overhead)가 발생함</li> <li>• 파일의 예비(Backup)와 회복(Recovery)이 어려움</li> <li>• 시스템이 복잡함</li> </ul>

#### \* 논리적 독립성과 물리적 독립성

- 논리적 독립성 : 응용 프로그램과 데이터베이스를 독립시킴으로써, 데이터의 논리적 구조를 변경시키더라도 응용프로그램은 변경되지 않음
- 물리적 독립성 : 응용 프로그램과 보조기억장치 같은 물리적 장치를 독립시킴으로써, 데이터베이스 시스템의 성능 향상을 위해 새로운 디스크를 도입하더라도 응용 프로그램에는 영향을 주지 않고 데이터의 물리적 구조만을 변경함

## ❖ 스키마(Database/DB)

### ❖ 정 의 :

- ◆ 데이터베이스의 구조와 제약조건에 관한 전반적인 명세(Specification)를 기술(Description)
- ◆ 데이터베이스를 구성하는  
데이터 개체(Entity), 속성(Attribute), 관계(Relationship) 및  
데이터 조작 시 데이터 값들이 갖는 제약조건등에 관해 전반적으로 정의
- ◆ 스키마는 사용자의 관점에 따라  
외부(External) 스키마, 개념(Conceptual) 스키마, 내부(Internal) 스키마로 나뉜다.
- ◆ 스키마(Schema)는 데이터 사전에 저장되며,  
다른 이름으로 메타 데이터(Meta-data)라고도 한다.

\* 메타 데이터(영어: database, DB)란? :

데이터(data)에 대한 데이터이다.

데이터에 관한 구조화된 데이터로, 다른 데이터를 설명해 주는 데이터이다.

위키백과 : <https://ko.wikipedia.org/wiki/메타데이터>

## ❖ 스키마(Database/DB)

### ❖ 스키마의 3 계층

#### ◆ 외부 스키마(External Schema) = 서브 스키마 = 사용자 뷰(View)

- 사용자나 응용 프로그래머가 각 개인의 입장에서 필요로 하는 데이터베이스의 논리적 구조를 정의
- 전체 데이터베이스의 한 논리적인 부분으로 볼 수 있으므로 서브 스키마(subschema)라고도 한다.
- 하나의 데이터베이스 시스템에는 여러 개의 외부 스키마가 존재할 수 있으며, 하나의 외부 스키마를 여러 개의 응용 프로그램이나 사용자가 공유할 수 있다.
- 같은 데이터베이스에 대해서도 서로 다른 관점을 정의할 수 있도록 허용한다.
- 일반 사용자는 질의어(SQL)를 사용하여 DB를 사용한다.

#### ◆ 개념 스키마 (Conceptual Schema) = 전체적인 뷰(View)

- 데이터베이스의 전체적인 논리적 구조로서, 모든 응용 프로그램이나 사용자들이 필요로 하는 데이터를 통합한 조직 전체의 데이터베이스로 하나만 존재한다.
- 개념 스키마는 개체 간의 관계와 제약조건을 나타내고 데이터베이스의 접근 권한, 보안 및 무결성 규칙에 관한 명세를 정의한다.
- 단순히 스키마(Schema)라고 하면 개념 스키마를 의미한다.
- 기관이나 조직체의 관점에서 데이터베이스를 정의한 것이다.
- 데이터베이스 관리자에 의해서 구성된다.

## ❖ 스키마(Database/DB)

### ❖ 스키마의 3 계층

#### ◆ 내부 스키마(Internal Schema)

- 물리적 저장장치의 입장에서 본 데이터베이스 구조
- 실제로 데이터베이스에 저장될 레코드의 물리적인 구조를 정의하고, 저장 데이터 항목의 표현방법, 내부 레코드의 물리적 순서 등을 나타낸다.
- 시스템 프로그래머나 시스템 설계자가 보는 관점의 스키마이다.
- 데이터베이스의 물리적 구조를 정의한다.
- 데이터의 실제 저장 방법을 기술한다.
- 물리적인 저장장치와 밀접한 계층이다.

## ❖ SQL(Structured Query Language)

### ❖ SQL이란?

- ◆ 구조화 질의어
- ◆ 관계형 데이터베이스 관리 시스템(RDBMS)의 데이터를 관리하기 위해 설계된 특수 목적의 프로그래밍 언어이다.
- ◆ 관계형 데이터베이스 관리 시스템에서 자료의 검색과 관리, 데이터베이스 스키마 생성과 수정, 데이터베이스 객체 접근 조정 관리를 위해 고안
- ◆ 많은 수의 데이터베이스 관련 프로그램들이 SQL을 표준으로 채택하고 있다.

### ❖ 역 사

- ◆ IBM에서 1970년대 초에 **도널드 D. 챔벌린**과 **레이먼드 F. 보이스**가 처음 개발하였다.  
초기에는 SEQUEL(Structured English Query Language, 구조 영어 질의어)라는 이름으로 시작하였으며, IBM의 준 관계형 데이터베이스 관리 시스템 **시스템 R**에 저장된 데이터를 조작하고 수신하기 위해 고안되었다.

## ❖ SQL(Structured Query Language)

### ❖ SQL 문법의 종류

- ◆ 데이터 정의 언어(DDL : Data Definition Language)
  - CREATE(데이터베이스 개체(테이블, 인덱스, 제약조건 등)의 정의)
  - DROP(데이터베이스 개체 삭제)
  - ALTER(데이터베이스 개체 정의 명령)
- ◆ 데이터 조작 언어(DML : Data Manipulation Language)
  - SELECT(테이블 데이터의 검색 결과 집합의 취득)
  - INSERT(행 데이터 또는 테이블 데이터의 삽입)
  - UPDATE(표 업데이트)
  - DELETE(테이블에서 특정 행 삭제)
- ◆ 데이터 제어 언어(DCL : Data Control Language)
  - GRANT(특정 데이터베이스 사용자에게 특정 작업을 수행 권한을 부여)
  - SET TRANSACTION(특정 데이터베이스 이용자로부터 이미 준 권한을 박탈)
  - BEGIN(트랜잭션 시작)
  - COMMIT(트랜잭션의 실행/작업 결과 테이블에 영구 반영)
  - ROLLBACK(트랜잭션 취소)
  - SAVEPOINT(무작위로 롤백 지점을 설정)
  - LOCK(표 등의 자원을 차지)