



CONTENTS

학습목표

- ◆ 데이터베이스란 무엇인지 설명할 수 있다.
- ◆ 데이터베이스의 물리적 구조를 설명할 수 있다.
- ◆ 데이터베이스 모델의 종류를 나열할 수 있다.
- ◆ 데이터베이스 관리 시스템에 대해 설명할 수 있다.





LEARNING

지난 주차 복습

05주차 학습내용. 운영체제

- 1 운영체제의 개요
- 2 운영체제 관리
- 3 운영체제의 분류
- 4 운영체제의 종류

지난 주차 **복습**

복습하기

운영체제의 개요

- 운영체제의 개념
 - 시스템의 하드웨어를 제어하고 사용자 및 응용 소프트웨어와 상호작용하는 시스템 소프트웨어의 일종
- 운영체제의 사용자 인터페이스 방식
 - CLI, GUI

운영체제 관리

- 기억장치 관리, 프로세스 관리, 입출력장치 관리, 파일 관리

지난 주차 **복습**

복습하기

운영체제의 분류

- 시분할 시스템, 다중 프로그램 시스템, 다중 처리 시스템, 다중 작업, 실시간 처리 시스템, 분산처리 시스템 등

운영체제의 종류

- 도스, 윈도우(GUI 방식), 매킨토시 운영체제, 유닉스(Unix), 리눅스(Linux) 등



1] 데이터와 정보

데이터 (Data)

단순한 사실에 불과한 아직 처리되지 않은 값

정보 (Information)

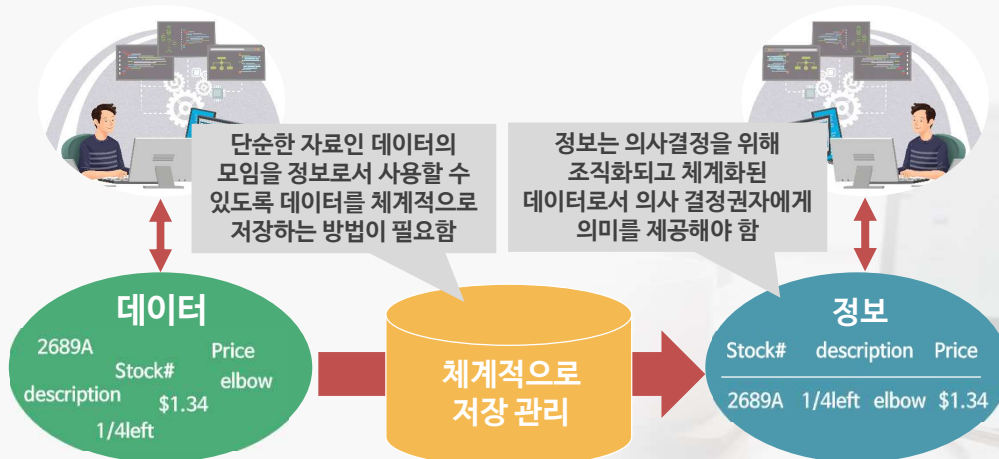
데이터가 사람에게 유용한 의미로 쓰여질 수 있도록 처리된 것

A small icon with the text '학습하기' (Study) next to it, located in the top right corner of the slide content area.

1] 데이터와 정보

학습하기

◆ 데이터에서 정보로 활용하기 위한 체계적 저장 관리

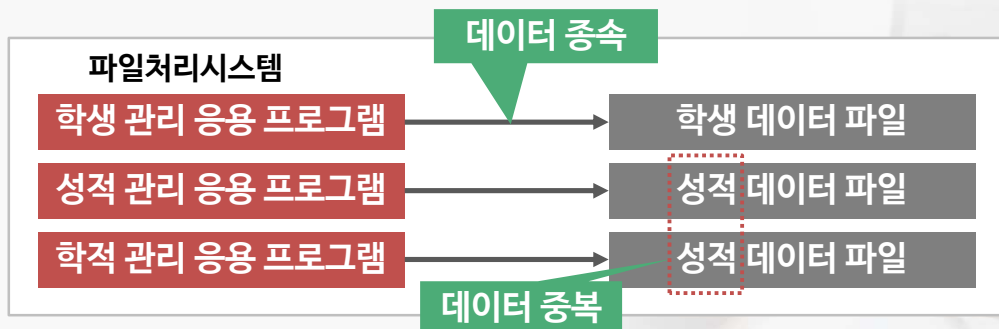


2] 파일처리시스템

학습하기

각 응용 프로그램마다 별도의 데이터 파일을 유지

파일처리시스템의 문제점



3] 데이터베이스의 정의 및 특징

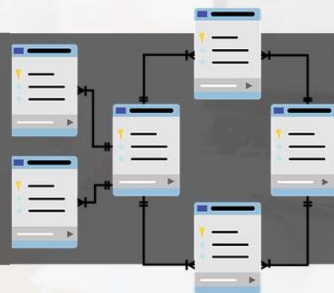
학습하기

◇ 데이터베이스의 정의

데이터베이스

- 관련 있는 데이터의 저장소
- 여러 사람이나 응용시스템에 의해 참조 가능하도록 **서로 논리적으로 연관되어 통합 관리되는 데이터의 모임**

데이터베이스에 저장된 자료는 데이터를 추가하고, 공유하고, 찾고, 정렬하고, 분류하고, 요약하고, 출력하는 등의 여러 조작을 통하여 정보로 활용될 수 있음



3] 데이터베이스의 정의 및 특징

학습하기

◇ 데이터베이스의 특징

주요 특징	내용
통합된 데이터	데이터의 특성, 실제 상호간의 의미 관계와 형식 관계를 기술한 개념적인 구조에 따라서 편성된 데이터의 집합
관련 있는 데이터	동시에 복수의 적용 업무나 응용 시스템에 대한 데이터의 공급기로서 공유할 필요가 있는 데이터를 보관, 관리
중복의 최소화	동일한 내용의 데이터가 중복되어 있지 않아야 하고, 다양한 접근 방식이 마련되어 있어야 하며, 검색이나 갱신이 효율적으로 이루어질 수 있도록 중복을 최소화
보조기억장치에 저장	자기 디스크나 자기 테이프 등 컴퓨터에서 사용할 수 있는 보조기억 장치에 저장
무결성	데이터가 정확성을 항상 유지
동시접근	여러 사람이 동시에 자료에 접근하더라도 문제없이 작업을 수행
보안 유지	데이터베이스의 관리 및 접근을 효율적으로 관리하여 보안 유지
장애 회복	문제가 발생하더라도 이전 상태로 복구 가능

4] 데이터베이스 관리 시스템

학습하기

◇ DBMS(Database Management System)의 정의

DBMS

- 데이터베이스를 관리하는 소프트웨어
- 사용자가 데이터베이스를 만들고, 유지 관리할 수 있도록 돕는 프로그램
- 데이터와 응용 프로그램 사이에서 중재자 역할로서 모든 프로그램들이 데이터베이스를 유용하게 활용할 수 있도록 관리해 주는 소프트웨어

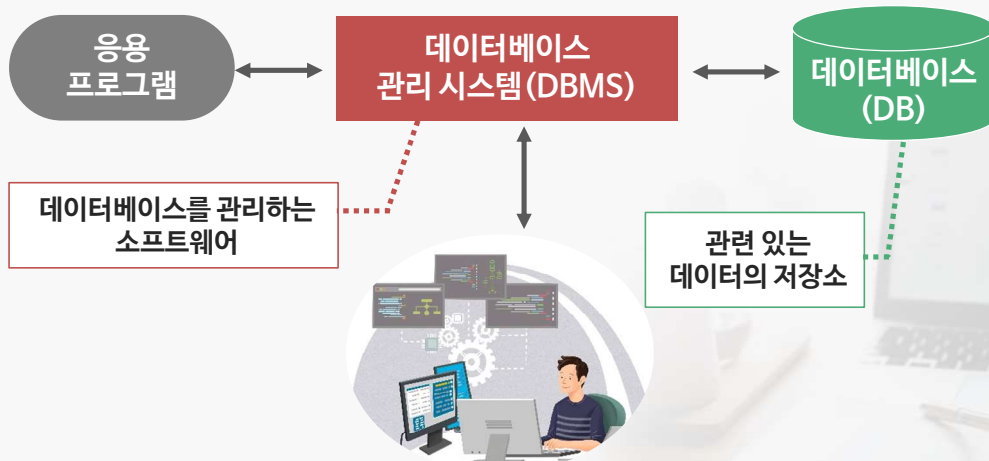
DBMS를 통한 데이터베이스 관리



4] 데이터베이스 관리 시스템

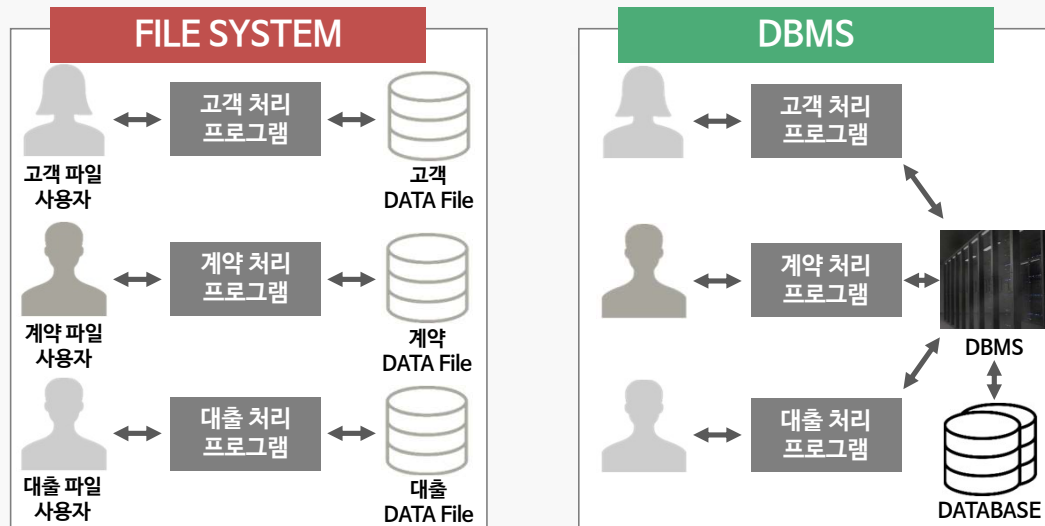
학습하기

◇ DBMS(Database Management System)의 정의



5] 파일 시스템 vs. DBMS 비교

학습하기



2

데이터베이스의 물리적 구조

1) 필드, 레코드, 파일

학습하기

필드 (Fields)

- 논리적인 의미 있는 자료의 단위
- **특정한 종류**의 데이터를 저장하기 위한 영역
 - 특정한 종류 : 그 필드에 저장될 수 있는 데이터의 종류로 데이터 유형(Data Type)

레코드 (Record)

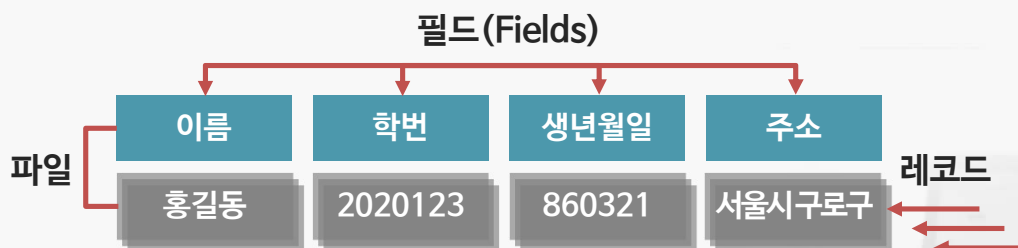
- 필드에는 실제 자료 값이 저장되며 이러한 필드가 여러 개 모이면 하나의 레코드(Record)가 됨

파일 (File)

- 레코드가 여러 개 모이면 하나의 파일이 됨

1) 필드, 레코드, 파일

학습하기



2] 파일과 데이터베이스

학습하기

파일

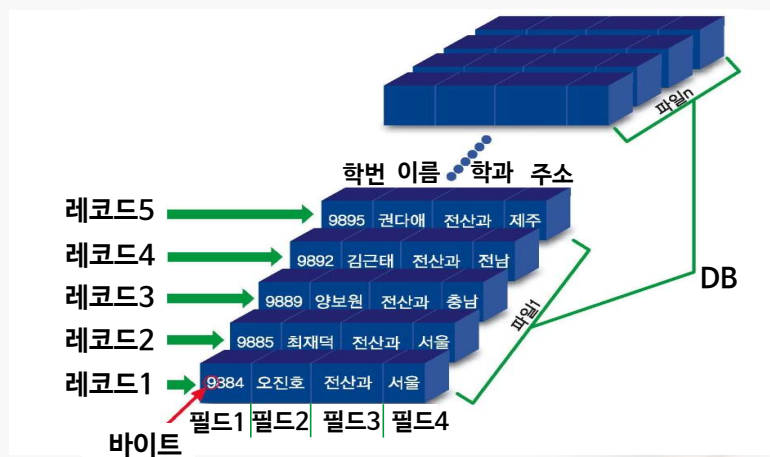
여러 개의 레코드가 모여 하나의 파일을 구성

데이터베이스 (Database)

파일을 여러 개 모아 저장된 파일들을 논리적으로 연결해서
필요한 정보를 적절히 활용할 수 있도록 서로 관련 있는
데이터들로 통합한 파일의 집합

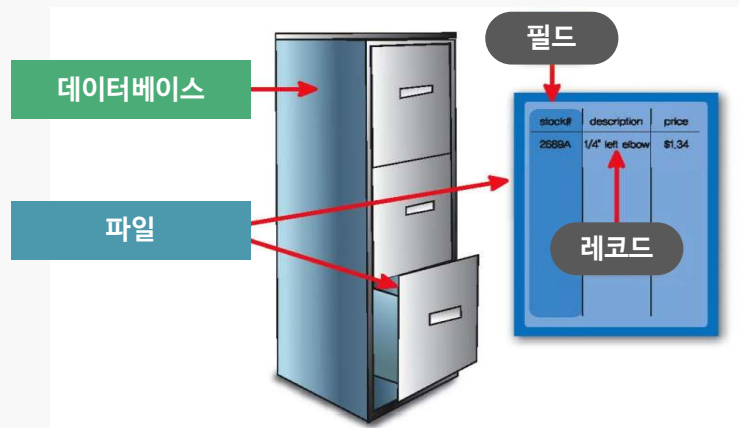
2] 파일과 데이터베이스

학습하기



3] 캐비닛과 데이터베이스

학습하기



3

데이터베이스 모델

1] 정의

학습하기

데이터베이스 모델

데이터의 논리적 설계와 그들 간의 관계를 표현



2] 데이터베이스 모델의 종류

학습하기

계층 모델 (Hierarchical Model)

네트워크 모델 (Network Model)

관계형 모델 (Relational Model)

객체-관계형 모델 (Object-Relational Model)

2] 데이터베이스 모델의 종류

학습하기

◇ 계층 모델



데이터는 위에서 아래로 트리 형태로 구성

- 각 엔터티는 하나의 부모만 가지며, 한 부모는 열 자식을 가질 수 있음



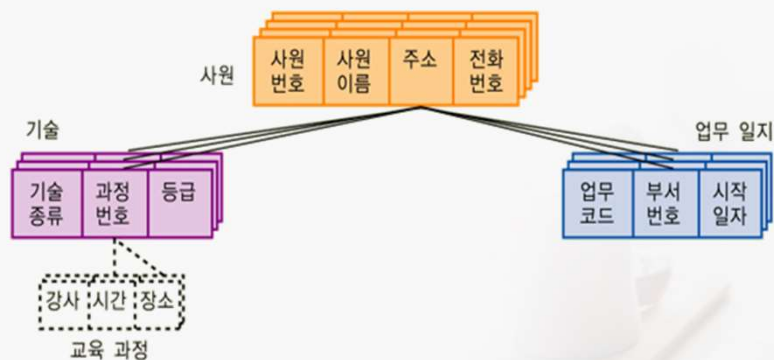
링크라는 개념에서는 네트워크 모델과 유사하나 레코드들이 트리 형태로 계층을 갖는 것이 다름

2] 데이터베이스 모델의 종류

학습하기

◇ 계층 모델

개인 신상의 계층형 데이터 모델



2] 데이터베이스 모델의 종류

학습하기

◇ 네트워크 모델

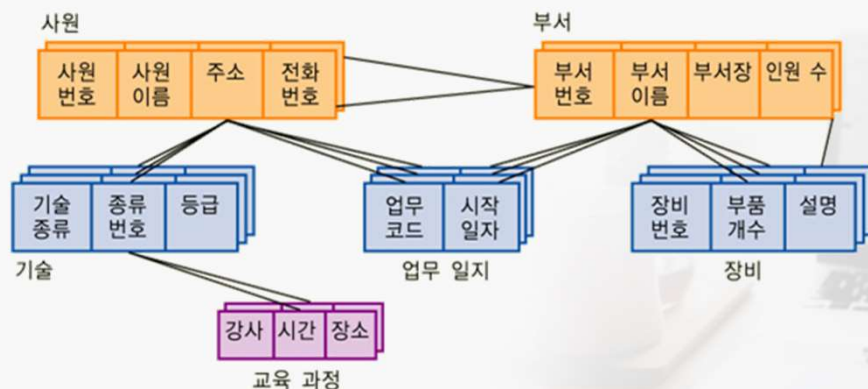
- ✓ 레코드와 레코드 간의 관계를 서로 연결하는 그래프를 사용한 모델
- ✓ 데이터 간의 관계는 링크(Link)로 표현
- ✓ 1971년 데이터 모델의 제안
- ✓ 트리 형식에서 벗어나 2개 이상의 부모 레코드를 허용

2] 데이터베이스 모델의 종류

학습하기

◇ 네트워크 모델

개인 신상의 네트워크형 데이터 모델



2] 데이터베이스 모델의 종류

학습하기

◇ 관계형 모델

- ✓ 데이터를 행과 열의 이차원 테이블의 집합으로 표현한 모델
- ✓ 포인터가 존재하지 않고 테이블을 구성하는 동일한 열로 데이터의 관계를 표현
- ✓ 1970년 에드가 코드에 의해 제한, 1980년대 이후 많이 쓰임
- ✓ 레코드는 키(key)단위로 구성
 - 데이터 간 연결은 키(key)의 중복으로 생성

2] 데이터베이스 모델의 종류

학습하기

◇ 관계형 모델

개인 신상의 관계형 데이터 모델 표현



2] 데이터베이스 모델의 종류

학습하기

◇ 객체-관계형 모델

- ✓ 클래스도 도메인이 될 수 있음
- ✓ 클래스의 한 속성값이 한 개 이상 존재
- ✓ 클래스는 메소드(method)를 가짐
- ✓ 계층적 클래스 구조를 가짐

2] 데이터베이스 모델의 종류

학습하기

◇ 객체-관계형 모델

개인 신상의 객체-관계형 데이터 모델 표현



3] 관계형 모델

학습하기

◇ 관계형 구조



관계형 모델은 데이터를 행과 열로 구성된 이차원 **테이블**의 집합으로 표현한 모델



관계(relation)



테이블 내의 필드 중에서 일부를 다른 테이블의 필드와 중복함으로써 여러 테이블 간의 상관 관계를 정의

3] 관계형 모델

학습하기

◇ 관계의 구성

예

관계 스키마와 관계 사례의 예

학생				
학번	이름	학과	주소	지도교수
2000001	오진호	001	서울	0001
2000002	권다애	002	경기도	0015
2000003	김근태	001	인천	0002
2000004	양보원	003	대전	0022
2000005	김태수	001	서울	0003

관계 스키마

관계의 구조 정의

관계 사례

관계 스키마에
삽입되는
실제 데이터 값

3] 관계형 모델

학습하기

◇ 관계의 구성 요소

Attributes (Columns)				
	A1	A2	An
Tuples (Rows)	t1			
	t2			
	tn			
	Domain D1	Domain D2		Domain D3

3] 관계형 모델

학습하기

◇ 관계의 구성 요소

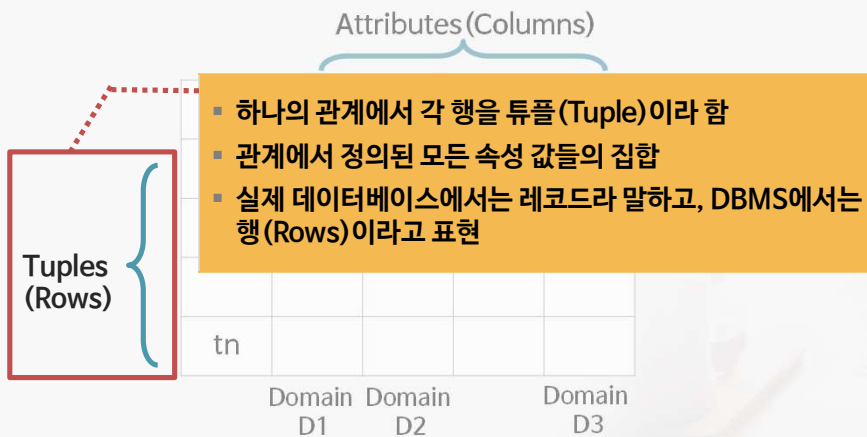
- 관계에서 각 열을 속성 (Attribute)이라 함
- 속성은 실제 데이터베이스에서는 필드라 말하고, DBMS에서는 열 (Columns)이라고 표현
- 한 테이블에서 유일한 이름을 가져야 함
- 한 관계의 총 속성의 수를 관계의 차수 (Degree)라 함
- 관계의 각 속성은 각 열에 저장되는 자료의 의미를 나타냄

Domain D1 Domain D2 Domain D3

3] 관계형 모델

학습하기

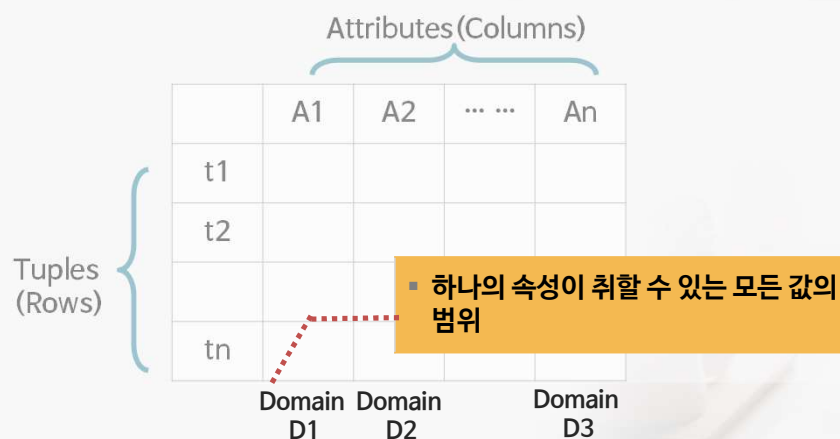
◇ 관계의 구성 요소



3] 관계형 모델

학습하기

◇ 관계의 구성 요소



3] 관계형 모델

학습하기

◇ 관계의 구성 요소

● 관계의 특징

“ 중복된 튜플은 삽입될 수 없으며,
튜플 내의 모든 값은 더 이상 나눌 수 없는 원자 값이어야 함 ”

특징	내용
속성 이름의 유일성	한 관계에서 속성이름은 유일해야 함
원자값	튜플 내의 모든 값은 더 이상 나눌 수 없는 원자값이어야 함
튜플 간의 무순서	관계에서 튜플 간의 순서는 무의미함
속성 간의 무순서	한 관계에서 속성 간의 순서는 무의미함
중복 불허	한 관계에서 두 튜플의 속성 값이 모두 같은 것은 불허함

3] 관계형 모델

학습하기

◇ 관계의 구성 요소

- 키(Key) : 관계에서 튜플들을 유일(uniqueness)하게 구별할 수 있는 하나 이상의 속성의 집합

후보키 (Candidate key)

- 하나의 관계에서 유일성과 최소성(Minimality)을 만족하는 키

주키 (Primary key)

- 관계에서 여러 튜플 중에서 하나의 튜플을 식별하는 역할을 수행

외래키 (Foreign key)

- 어느 관계의 속성들 중에서 일부가 다른 관계의 주키가 될 때의 키
- 관계와 관계를 서로 연결할 수 있음

3] 관계형 모델

학습하기

◇ 관계의 구성 요소

키(Key)

주키

학생

학번	이름	학과	주소	지도교수
2020001	오진호	001	서울	0001
2020002	권다애	002	경기도	0015
2020003	김근태	001	인천	0002
2020004	양보원	003	대전	0022
2020005	김태수	001	서울	0003

외래키

교수

교수번호	이름	연구실	전화번호
0001	김수철	T102	4444
0002	이기문	T103	4445
0003	박동진	T302	5555
....

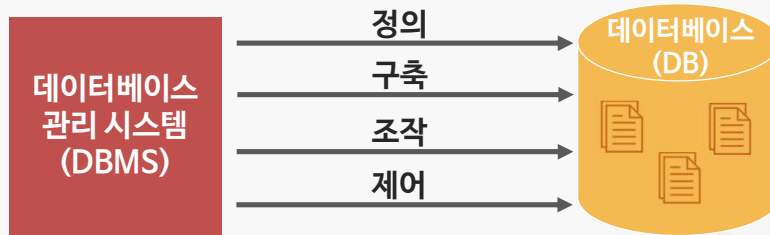
4

데이터베이스 관리 시스템

1] DBMS 정의

학습하기

◆ 데이터베이스 관리시스템 (DBMS : Database Management System)



- ✓ 데이터베이스에서 정보를 쉽게 활용할 수 있도록 만든 프로그램이자 소프트웨어
- ✓ 데이터베이스를 관리하는 소프트웨어

2] DBMS의 구성

학습하기

1 DBMS 인터페이스 도구 (DBMS Interface Tool)

- 사용자가 DB의 자료와 DB의 모든 장치에 쉽게 접근할 수 있는 수단을 제공하는 프로그램

2 질의 처리기 (Query Processor)

- 사용자와 응용프로그램이 요청하는 질의문을 해석하여 최적의 결과를 도출하는 구성요소

2] DBMS의 구성

학습하기

3 저장 관리자(Storage Manager)

- 데이터베이스 구조를 결정하고 최적의 데이터베이스를 구축하기 위하여 실제 저장장치에 자료를 저장하는 구성요소

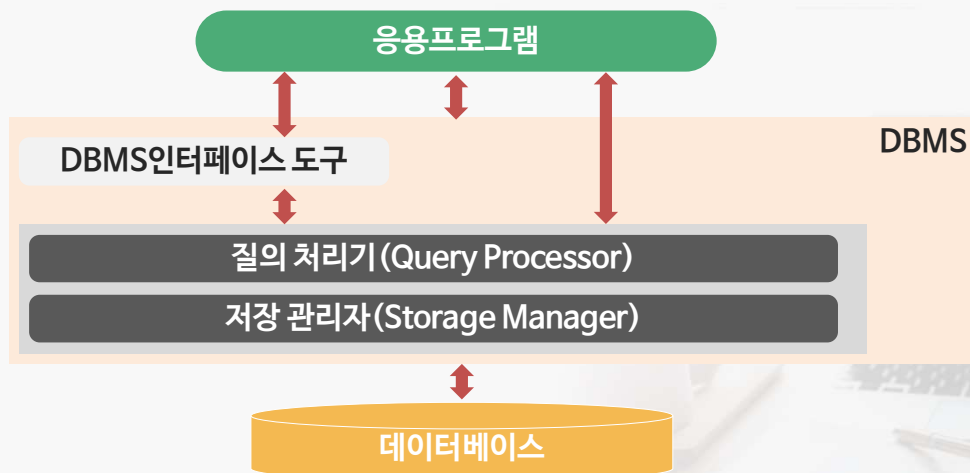
4 데이터베이스 시스템(Database System)

- DBMS + 응용 프로그램 + 실제 데이터
- 특정한 목적을 위하여 DBMS와 데이터베이스 그리고 이를 사용하는 응용 프로그램이 통합된 시스템

2] DBMS의 구성

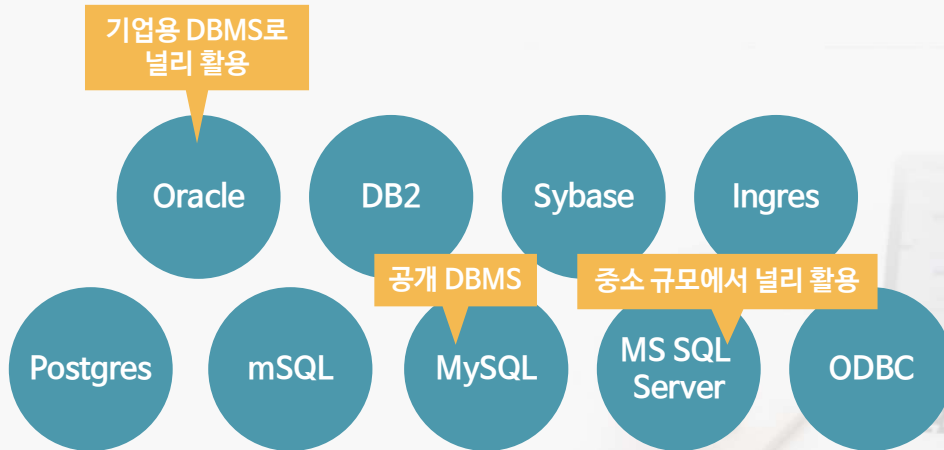
학습하기

◇ 데이터베이스의 구성요소와 데이터베이스 시스템



3] DBMS의 종류

학습하기



3] DBMS의 종류

학습하기



3] DBMS의 종류

학습하기

◆ 데이터베이스 관리시스템(DBMS) 현황

기업	대표 솔루션	특징
오라클	오라클DB	유닉스 환경 관계형DBMS(RDB)
티맥스데이터	티베로	우너천기술을 확보한 국산 DBMS
아마존웹서비스(AWS)	오로라 DB	상용 · 오픈소스 효율성 결합 RDB
마이크로소프트(MS)	SQL 서버	하이브리드형 데이터 플랫폼
IBM	DB2	1983년 발표된 상업용 RDB
구글 클라우드	클라우드 SQL	클라우드 네이티브 RDB
큐브리드	큐브리드 DB	오픈소스 기반 국산 RDB
마리아DB	마리아DB	오픈소스 기반 외산 RDB
몽고DB	몽고DB	오픈소스 기반 외산 NoSQL DB

3] DBMS의 종류

학습하기

1 오라클

ORACLE®

- 1977년 설립된 오라클(Oracle)사가 개발한 오라클은 세계적으로 가장 성공한 DBMS의 한 제품
- 기업용 대규모 데이터베이스 시스템의 DBMS로 자리잡음

▪ 오라클 에디션

Enterprise
Edition

Standard
Edition

Standard
Edition One

Personal
Edition

Lite Edition

3] DBMS의 종류

학습하기

2 MySQL



- 대표적인 오픈 소스 DBMS 제품
 - www.mysql.com에서 다운로드 가능
 - 구분 : 공개용, 상업용됨

- MySQL은 원래 mSQL이라는 DBMS에서 기반이 되어 새로 개발된 DBMS

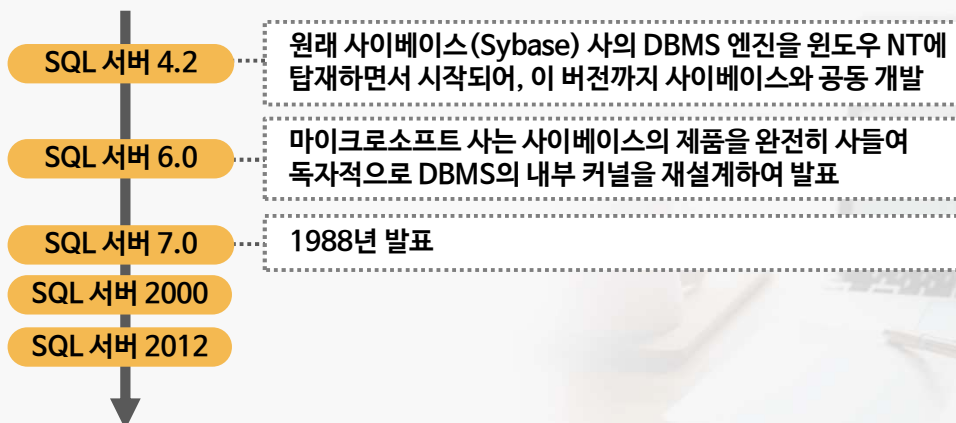


3] DBMS의 종류

학습하기

3 SQL 서버

- 마이크로소프트 사의 SQL 서버(SQL Server)는 인텔 기반의 서버용 컴퓨터에서 널리 사용되는 DBMS



3] DBMS의 종류

학습하기

4 ODBC(Open Database Connectivity)

- 데이터베이스를 사용하기 위한 **표준 개방형 응용 프로그램**
- DBMS의 종류에 관계없이 어떤 응용 프로그램에서나 모두 접근하여 사용할 수 있도록 하기 위하여 마이크로소프트에서 개발한 **데이터베이스 표준 접근 방법**

장점

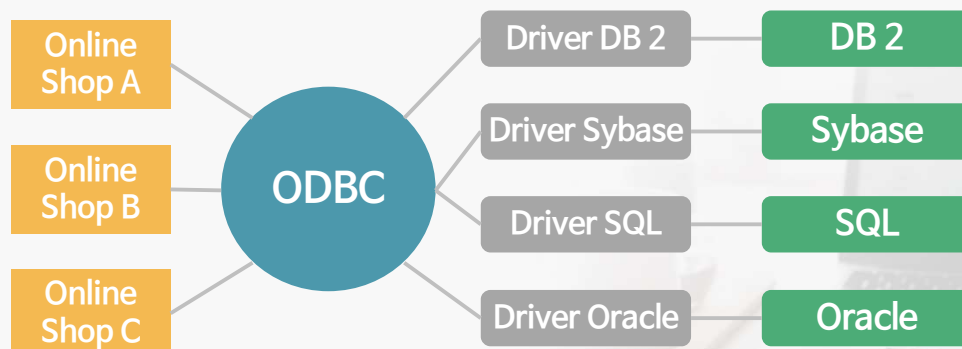
- 마이크로소프트는 DBMS를 위한 미들웨어인 ODBC(Open Database Connectivity) API(Application Programming Interface)를 제공하여 데이터베이스 개발을 더욱 쉽게 가능하게 함
- ODBC를 사용하면 여러 종류의 데이터베이스를 함께 사용할 수 있고 기존에 사용하던 데이터베이스를 교체한다 하더라도 응용 시스템을 계속해서 그대로 사용할 수 있어 비용을 절감할 수 있음

3] DBMS의 종류

학습하기

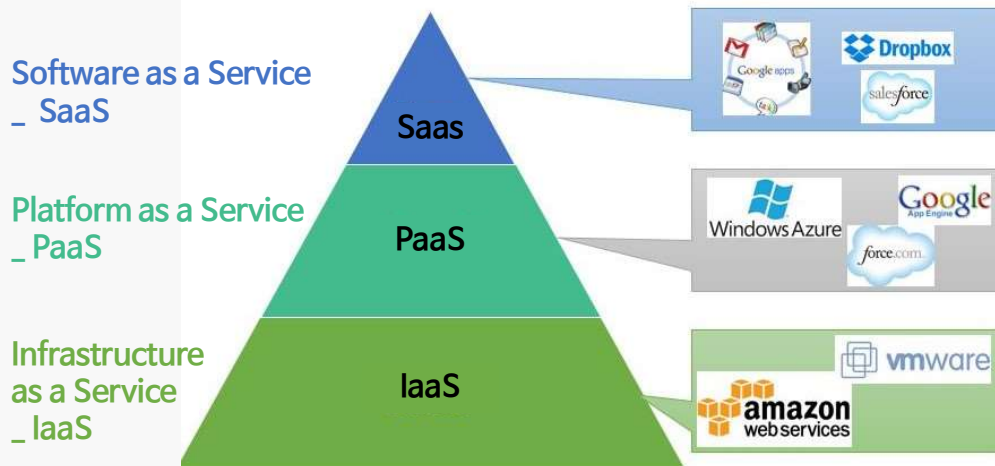
4 ODBC(Open Database Connectivity)

개념도



4] 클라우드 서비스

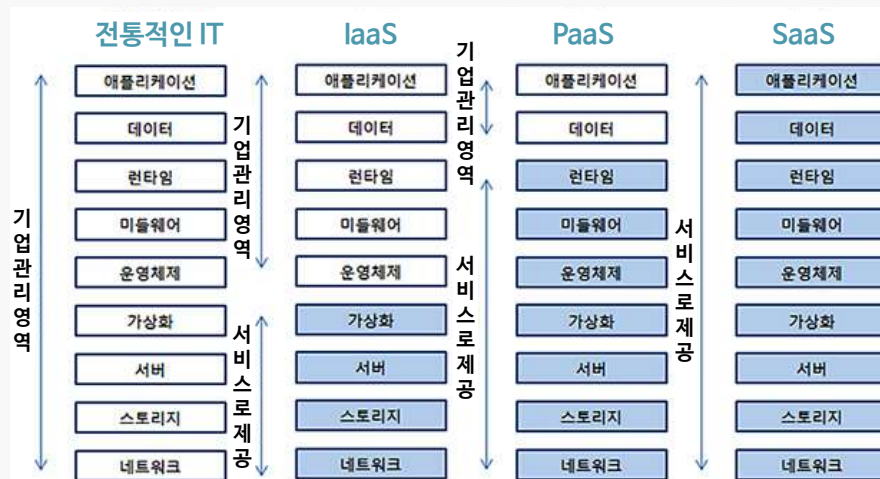
학습하기



4] 클라우드 서비스

학습하기

◇ 클라우드 서비스 모델 비교



SUMMARY 정리하기

정리하기

데이터베이스의 개요

- ◆ 데이터베이스의 정의
 - 관련 있는 데이터의 저장소로 서로 논리적으로 연관되어 통합 관리되는 데이터의 모임
- ◆ 데이터베이스의 특징
 - 통합된 데이터, 중복을 최소화, 무결성, 동시 접근, 보안 유지, 장애 회복 등



SUMMARY 정리하기

정리하기

데이터베이스의 물리적 구조


- ◆ 필드
- ◆ 레코드
- ◆ 파일
- ◆ 데이터베이스



SUMMARY
정리하기

데이터베이스 모델


- 계층 모델
- 네트워크 모델
- 관계형 모델
 - 모든 데이터를 이차원의 테이블로 표현한 모델
 - 관계는 관계 스키마와 관계 사례로 구성
 - 관계의 구성요소: 속성, 튜플, 키
- 객체-관계형 모델



SUMMARY
정리하기

데이터베이스 관리 시스템

- 정의
 - 데이터베이스를 관리하는 소프트웨어객체-관계형 모델
- 구성
 - 저장 관리자, 질의 처리기, DBMS 인터페이스 도구
- 종류
 - 오라클, DB2, MySql, SQL Server, Access 등



ANNOUNCEMENT

차시예고

정리하기

8주차

9주차

10주차

11주차

12주차

13주차

14주차

기말고사

컴퓨터 네트워크

- 수고하셨습니다.