**2과목 . 데이터 베이스 실무 응용**

**데이터베이스** : 특정 조직의 업무를 수행하는 데 필요한 상화 관련된 데이터들의 모임

- **통합 데이터** : 검색의 효율성을 위해 중복이 최소화된 데이터의 모임

- **저장 데이터** : 컴퓨터가 접근 가능한 저장 매체에 저장된 데이터

- **운영 데이터** : 조직의 목적을 위해 존재 가치가 확실하고 반드시 필요한 데이터

- **공유 데이터** : 여러 응용 프로그램들이 공동으로 사용하는 데이터

**특징**

- **실시간 접근성** : 사용자의 질의에 대하여 즉시 처리하여 응답하는 특징

- **계속적인 진화** : 삽입, 삭제, 갱신을 통하여 항상 최근의 정확한 데이터를 동적으로 유지하는 특징

- **동시 공유** : 여러 사용자가 동시에 원하는 데이터를 공용할 수 있는 특징

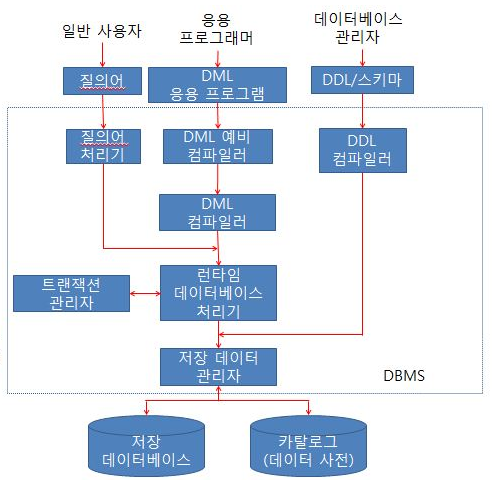
- **내용에 의한 참조** : 데이터베이스에 있는 데이터를 참조할 때 튜플의 주소나 위치에 의해서가 아니라

사용자가 요구하는 데이터 내용에 따라 참조하는 특징

- **데이터의 논리적, 물리적 독립성**

**데이터 베이스 시스템의 구성요소**

- 데이터 베이스, 스키마, DBMS, 데이터베이스 언어, 데이터베이스 컴퓨터, 데이터 베이스 사용자



**데이터 언어**

- DDL : Data Definition Language 데이터 정의어

- DCL : Data Control Language 데이터 제어어

- DML : Data Manipulation Language 데이터 조작어

**데이터 베이스 사용자**

- **데이터베이스 관리자** : 데이터 베이스를 정의하고 제어하는 사람

- **데이터 관리자** : 데이터에 대한 관리 총괄

- **데이터 설계자** : 데이터의 구조를 체계적으로 정의하는 사람

- **응용 프로그래머**

- **일반 사용자**

**DSMS** Data Stream Management System

- 대량의 데이터 스트림이라는 동적인 특성을 가진 데이터를 처리하고 관리하는 시스템

**스키마** : 데이터 베이스를 구성하는 데이터 개체, 속성, 관계 및 데이터 조작시 갖는 제약조건 등을 정의

-> 데이터 사전에 저장 된다

**데이터 사전** : 데이터베이스에 저장되어 있는 모든 데이터 개체들에 대한 정보를 유지, 관리하는 시스템

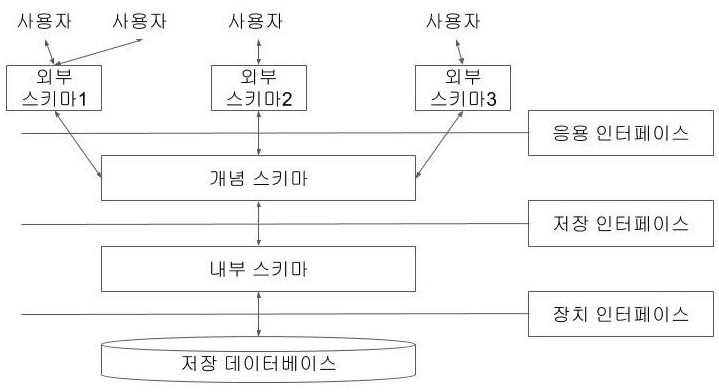
시스템 카탈로그라고도 한다.

**메타 데이터** : 데이터에 관한 데이터

**외부 스키마** = 논리 스키마 \_ 개인의 입장에서 필요로 하는 데이터베이스의 논리적 구조

**개념 스키마** = 기본적인 스키마 \_ 데이터 베이스의 명세 정의

**내부 스키마 \_** 데이터베이스의 물리적 구조



**데이터 베이스의 설계**

: 요구조건 분석 -> 개념적 설계 -> 논리적 설계 -> 물리적 설계 -> 데이터베이스 구현

**E-R 모델** : 개념적 데이터 모델 \_ **1976년 피터 첸에 의해 제안**

**릴레이션** : 데이터를 원자 값으로 갖는 이차원의 테이블

-> **릴레이션 스키마** : 구조를 나타낸다.

-> **릴레이션 인스턴스** : 실제 값

|  |
| --- |
| 속성 : Attribute / 속성의 개수 : Degree / 속성 값이 취할 수 있는 원자 값들의 집합 : Domain  튜플 : Tuple / 튜플의 개수 : 카디널리티 |

**키의 개념**

**1. 슈퍼키** : 한릴레이션 내에 있는 속성들의 집합으로 구성된 키

유일성은 만족/ 최소성은 만족 X

**2. 후보키** : 릴레이션을 구성하는 속성들 중 튜플을 유일하게 식별하기 위해 사용되는 속성들의 부분집합

유일성, 최소성 만족

**3. 기본키** : 후보키 중에서 특별히 선정된 키

**4. 대체키** : 후보키d서 기본키를 제외한 나머지 후보키 ( 후보키 – 기본키 )

**5. 외래키** : 다른 릴레이션의 기본키를 참조하는 속성

**무결성** : 데이터베이스에 저장된 데이터 값과 그것이 표현하는 현실 세계의 실제 값이 일치하는 정확성

**- NULL 무결성** : 릴레이션의 특정 속성 값이 NULL이 될 수 없도록 하는 규정

**- 고유 무결성** : 릴레이션의 특정 속성에 대해 각 튜플이 갖는 속성 값들이 서로 달라야 하는 규정

- **도메인 무결성** : 특정 속성의 값이 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정

**- 키 무결성** : 하나의 릴레이션에는 적어도 하나의 키가 존재해야 한다는 규정

**- 관계 무결성** : 릴레이션에 어느 한 튜플의 삽입 가능 여부 또는 한 릴레이션과 다른 릴레이션의 튜플들

사이의 관계에 대한 적절성 여부를 지정한 규정

**- 참조 무결성** : 외래키 값은 NULL이거나 참조 릴레이션의 기본키 값고 동일해야 한다는 규정

**- 개체 무결성**: 기본 릴레이션의 기본키를 구성하는 어떤 속성도 NULL일 수 없다는 규정

관계 대수 : 관계형 데이터베이스에서 원하는 정보와 그 정보를 어떻게 유도하는가를 기술하는 절차적 언어

관계 해석 : 비절차적 특성

순수 관계 연산자

**SELECT σ** : where

**PROJECT π.** : 속성 리스트

**JOIN ▷◁**  : 두개의 릴레이션을 합친다

**DIVISION ÷** :

일반 집합 연산자

**합집합 U** :

**교집합** ∩:

**차집합 –** :

**교차곱 X** :

DDL

- CREATE, ALTER, DROP

**스키마 정의**

**CREATE SCHEMA** 스키마이름 **AUTHORIZTION 사용자** ID;

도메인 정의

**CREATE DOMAIN** 도메인명 데이터 타입

[ **DEFAULT** 값 ]

[ **CONSTRAINT** 제약조건이름 **CHECK** 범위값 ];

테이블 정의

**CREATE TABLE** 테이블 이름 (

속성이름 데이터 타입 [ NOT NULL ],

[ **PRIMARY KEY** 속성이름 ]

[ **FOREIGN KEY** 속성이름

**REFERENCES** 테이블(기본 속성키)

**ON DELETE** 옵션 -> NO ACTION / CASCADE / SET NULL / SET DEFAULT

**ON UPDATE** 옵션 ],

[ **CONSTRAIN**T 제약조건명 **CHECK** 조건식 ];

**뷰 정의**

**CREATE VIEW** 뷰이름(속성이름, 속성이름 -- )

**AS SLECT문;**

**인덱스 정의**

CREATE [ UNIQUE ] INDEX 인덱스명

ON 테이블명(속성명 정렬옵션);

**트리거** : 데이터베이스 시스템에서 데이터의 입력, 갱신, 삭제 등의 이벤트가 발생할 때마다

자동적으로 수행되는 사용자 정의 프로시저

**트리거 정의**

**CREATE TREIGGER** 트리거명 [동작시기][동작 옵션]

**REFERENCING** [ NEW | OLD ] **TABLE AS** 테이블명

**FOR EACH NOW**

**WHEN** 조건식

트리거

**테이블 정의 변경**

**ALTER TABLE** 테이블명 **ADD** 속성명 데이터 타입 [DEFAULT 값]; 테이블 컬럼 추가

**ALTER TABLE** 테이블명 **ALTER** 속성명 [ SET DEFAULT 값 ]; 테이블 컬럼 기본값 변경

**ALTER TABLE** 테이블명 **DROP COLUMN** 속성명 [ CASCADE ]; 테이블 컬럼 삭제

**삭제**

**DROP** 해당 해당명 옵션

**SELECT**

**SELECT** [ DISTINCT 중복 제거] 속성명

**FROM** 테이블명

**WHERE** 조건

**GROUP BY** 속성명

**HAVING** 조건

**ORDER BY** 속성명 ASC/DESC

**삽입**

**INSERT INTO** 테이블명 **VALUES**;

**삭제**

**DELETE FROM** 테이블명 **WHERE** 조건;

**수정**

**UPDATE** 테이블명 **SET** 속성명= 데이터

**WHERE** 조건;

**COMMIT** : 트랜잭션의 모든 변경 내용들을 영구적으로 데이터베이스에 반영하는 명령어

**ROLLBACK** : 변경된 모든 내용들을 취소하고 데이터베이스를 이전 상태로 되돌리는 명령어

**GRANT**

**GRANT** 권한 **ON** 개체 **TO** 사용자 **WITH GRANT OPTION**

**REVOKE**

**REVOKE GRANT OPTION FOR** **ON** 개체 **FROM** 사용자;

**이상** : 테이블에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하고,

이 중복으로 인해 테이블 조작시 문제가 발생하는 현상

**삽입이상 / 삭제이상 / 갱신이상**

**함수적 종속**

어떤 테이블 R에서 X와 Y를 각각 R의 속성 집합의 부분 집합이라 하자.

속성 X의 값 각각에 대해 시간에 관계없이 항상 속성 Y의 값이 오직 하나만 연관되어 있을 때

Y는 x에 함수적 종속되었다고 한다. ㅠ

**정규화**

- 테이블의 속성들이 상호 종속적인 관계를 갖는 특성을 이용하여 테이블을 무손실 분해하는 과정

**제 1정규형**

테이블에 속한 모든 속성의 도메인이 원자값 만으로 되어 있는 정규형

**제 2 정규형**

기본키가 아닌 모든 속성이 기본키에 대하여 완전 함수적 종속을 만족하는 정규형

**제 3 정규형**

기본키가 아닌 모든 속성이 기본키에 대해 이행적 함수적 종속을 만족하지 않는 정규형

**BCNF**

모든 결정자가 후보키인 정규형

**도** 메인이 원자값

**부** 분함수 종속 제거

**이** 행적 함수 종속 제거

**결** 정자가 모두 후보키

**다** 중값 종속

**조** 인 종속 후보키 성립

**트랜잭션의 특성**

**1. 원자성** : 트랜잭션의 연산은 데이터베이스에 모두 반영되든지 아니면 전형 반영되지 않아야한다.

**2. 일관성** : 트랜잭션 수행이 성공적으로 완료되면 언제나 일관성 있는 데이터베이스 상태로 변환한다.

**3. 독립성** : 한 트랙잭션이 데이터를 갱신하는 동안 이 트랜잭션이 완료되기 전에는 갱신 중인 데이터를

다른 트랜잭션들이 접근하지 못하도록 해야 한다.

**4. 영속성** : 트랜잭션의 실행이 성공적으로 실행 완료된 후에는 시스템에 오류가 발생하더라도

트랜잭션에 의해 변경된 내용은 계속 보존되어야 한다.

**장애의 유형**

- 실행장애

- 트랜잭션 장애

- 시스템 장애

- 미디어 장애

**회복 관리기**

구성요소 : **로그, 메모리 덤프 -redo undo**

회복 기법

연기 갱신 / 즉각 갱신 / 그림자 페이지 / 검사점 / 미디어 회복

병행 제어

- 다중 프로그램의 이점을 활용하여 동시에 여러 개의 트랜잭션을 병행 수행할 때 실행되는 트랜잭션들이

데이터베이스의 일관성을 파괴하지 않도록 트랜잭션 간의 상호 작용을 제어하는 기술

튜닝

- 데이터베이스 응용 프로그램, 데이터베이스 자체, 운영체 등의 조정을 통하여

데이터베이스 관리 시스템의 성능을 향상시키는 작업

데이터 웨어 하우스

데이터 마트 : 소규모 데이터 웨어 하우스

데이터 마이닝 : 데이터 웨어하우스에 저장된 집합에서 사용자의 요구에 따라  
 유용하고 가능성 있는 정보를 발견하기 위한 기법

OLAP : 다차원으로 이루어진 데이터로부터 통계적인 요약 정보를 분석하여 의사 결정에 활용하는 방식

**3과목 . 업무 프로세스**

**ISP** Information Startegy Planning 정보전략 계획

기업이 수립한 장기적인 목표와 방향을 효과적으로 지원하기 위해 전사적 관점에서 정보시스템이나

정보 관리 등의 전략을 수립하는 활동

CSF Critical Success Factor