

## [데이터베이스-출제경향 분석]

### 1. 데이터베이스 강의 목차

### 2. 기출문제 출제경향 분석

|                          |                |  |
|--------------------------|----------------|--|
| 01강 - 데이터베이스 정의          | 3%             |  |
| 02강 - DBMS,스키마,데이터언어,DBA | 11%            |  |
| 03강 - DB설계,데이터모델,ER모델    | 15%            |  |
| 04강 - 논리적데이터모델,관계형DB     | 12%            |  |
| 05강 - 관계대수,관계해석,정규화      | 8%             |  |
| 06강 - SQL,뷰,시스템카탈로그      | 16%            |  |
| 07강 - 내장SQL,고급데이터베이스     | 15% (산업기사 4%)  |  |
| 08강 - 자료구조(선형/비선형구조)     | 12% (산업기사 18%) |  |
| 09강 - 자료구조(정렬/검색)        | 5% (산업기사 8%)   |  |
| 10강 - 자료구조(파일편성)         | 3%             |  |

**80%**

데이터베이스 설계 및 이용

↑

↓

데이터 저장 방법

**20%**

## [DB 1강-데이터베이스 정의]

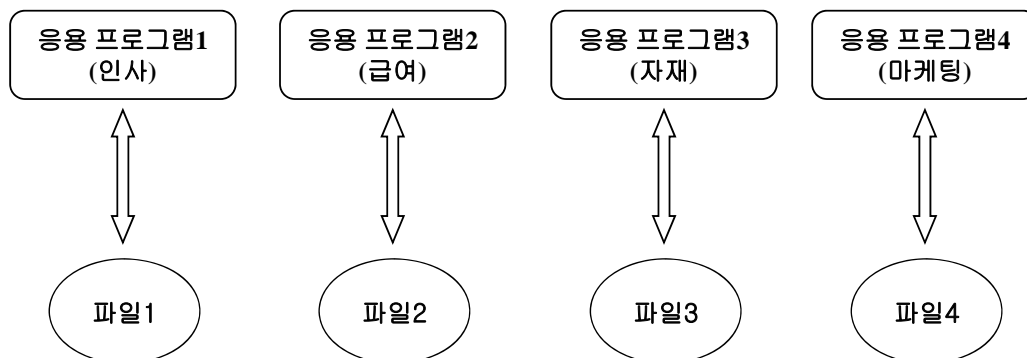
## [출제빈도 '하']

### 1. 데이터베이스 정의 ★★★★★

: 특정 조직이 업무 수행하는 데 필요한 관련성 있는 자료들의 집합체 (통합, 저장, 운영, 공용)

### 2. 데이터베이스 시스템 도입 배경

: 파일 시스템의 문제점을 해결



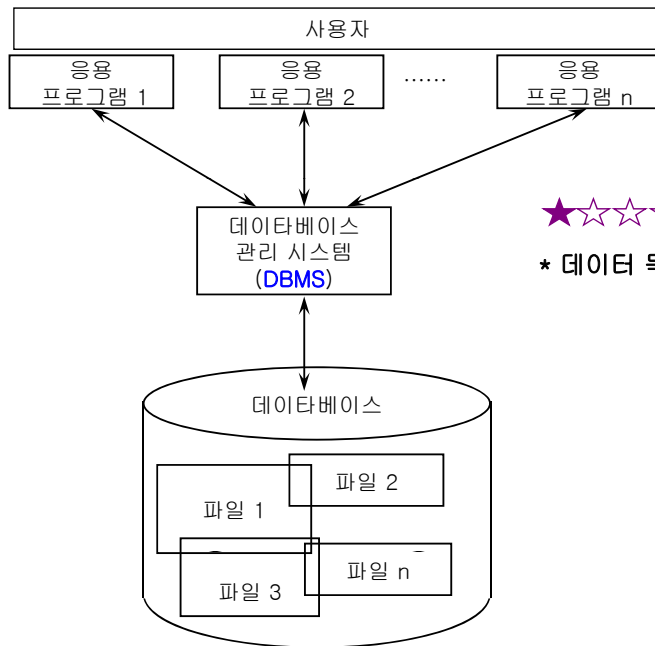
★★★★☆

\* 독립된 파일 단위로 업무와 관련한 데이터를 저장하므로 데이터 중복성과 데이터 종속성 발생

→ 데이터 무결성 위배 가능성 높음

## [DB 1강-데이터베이스 정의]

### 3. 데이터베이스 시스템 개념



☆☆☆☆☆  
\* 데이터 독립성

물리적 데이터 독립성 : 기존 응용 프로그램에 영향을 주지 않고 데이터의 물리적인 구조를 변경할 수 있는 것을 말한다.

논리적 데이터 독립성 : 데이터의 논리적 구조를 변경시키더라도 응용 프로그램은 변경되지 않는다.

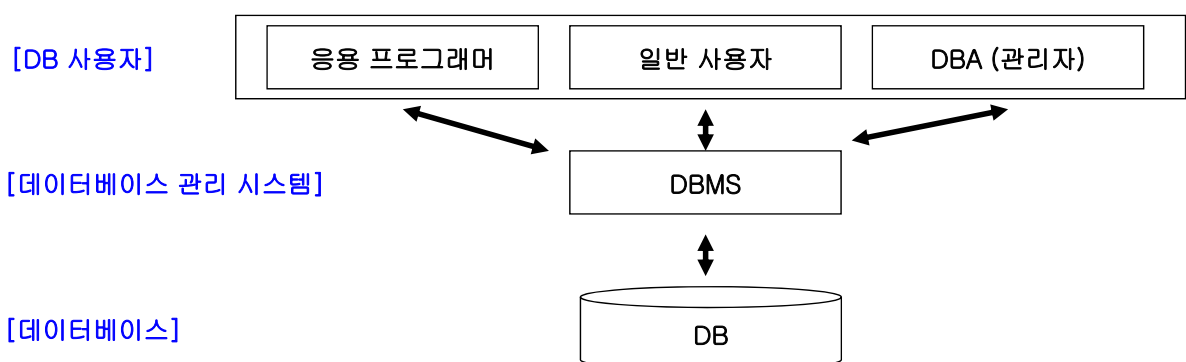
\* 데이터를 통합 운영하므로 **중복성 감소**, 불일치 감소 → 데이터 일관성, 무결성 유지 ☆☆☆☆☆

## [DB 1강-데이터베이스 정의]

### 4. 데이터베이스 특징 ☆☆☆☆☆

- 1) 실시간 접근 : 내가 원할 때마다 언제든지 바로 접근해서 자료를 처리할 수 있다.
  - 2) 계속적인 변화 : 데이터의 삽입, 삭제, 갱신 작업으로 항상 최신의 데이터를 유지해야 한다.
  - 3) 공용 : 여러 사용자가 같이 쓸 수 있어야 한다.
  - 4) 내용에 의한 참조 : 위치나 주소가 아닌 데이터의 내용, 즉 값에 따라 참조할 수 있다.
- 비용 감소 (X)

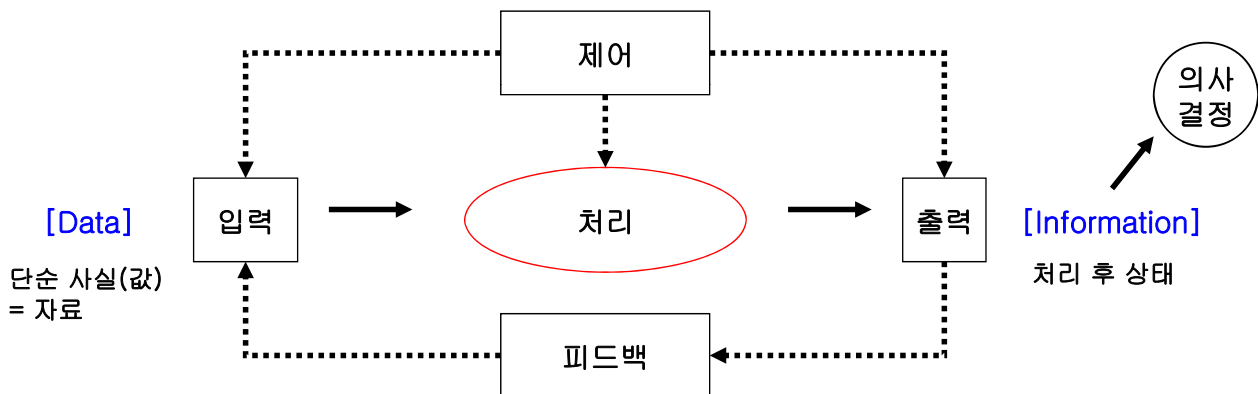
### 5. 데이터베이스 시스템 구성



## [DB 1강-데이터베이스 정의]

### 1. 정보 시스템 ★★☆☆☆

: 한 조직체의 데이터를 바탕으로 의사결정에 필요한 정보를 추출하고 생성하는 시스템



### 2. 자료처리 시스템

: 정보 시스템에서 처리 과정을 의미함

## [DB 1강-데이터베이스 정의]

### 3. 자료처리 시스템의 종류 ★☆☆☆☆

#### 1) 일괄 처리 시스템 (Batch Processing System)

- 일정시간 또는 일정량의 데이터를 한꺼번에 모아서 처리 (시스템 중심)
- 각 트랜잭션 당 처리비용이 적게 든다
- 예) 급여 계산, 회계 마감 업무, 세무 처리 등

#### 2) 온라인 실시간 처리 시스템 (Real-time Processing System)

- 데이터가 발생하는 즉시 처리하여 결과를 산출하도록 하는 시스템 (사용자 중심)
- 예) 기차 예매, 티켓 예매, 은행 업무 등

#### 3) 분산 처리 시스템

- 컴퓨터들이 지리적으로 분산되어 있지만, 실제 사용자들이 볼 때는 논리적으로 하나로 연결되어 있는 것처럼 보여져서 처리되는 시스템

### 4. 데이터웨어하우스(Datawarehouse)

: 기간 업무 시스템에서 추출되어 새로이 생성된 데이터베이스로서 의사결정자원 시스템을 지원하는 주체적, 통합적, 시간적 데이터의 집합체

## [DB 1강-데이터베이스 정의]

1. 한 조직체에 필요한 Data를 수집, 저장해 두었다가 필요 시에 처리해서 의사결정에 유용한 정보를 생성하고 분배하는 수단을 무엇이라 하는가?

- 가. 자료 처리 시스템                      나. 정보 시스템
- 다. 전문가 시스템                        라. 응용 시스템

2. 자료(Data)와 정보(Information)에 대한 설명이 가장 적절한 것은?

- 가. 정보란 자료를 처리해서 얻을 수 있는 결과이다.
- 나. 자료란 적절한 의사 결정의 수단으로 사용할 수 있는 시작이다.
- 다. 정보란 현실 세계에 존재하는 가공하지 않은 그대로의 모습을 의미한다.
- 라. 자료와 정보는 같은 의미이다.

3. 일괄 처리 방식을 적용한 업무 형태로서 부적합한 것은?

- 가. 급여 계산
- 나. 회계 마감업무
- 다. 세무 처리
- 라. 예약 업무

[정답] 1.나 2.가 3.라 4.라 5.다 6.라

4. 데이터베이스의 특성이 아닌 것은?

- 가. 실시간 접근성(Real-Time Accessibility)
- 나. 내용에 의한 참조(Content Reference)
- 다. 동시 공유(Concurrent Sharing)
- 라. 이산적 변화(Discrete Evolution)

5. 데이터베이스의 정의로 보기 어려운 것은?

- 가. 동일한 데이터의 중복을 최소화한다.
- 나. 컴퓨터가 접근할 수 있는 저장매체에 저장된 데이터의 집합이다.
- 다. 특정 프로그램을 위한 독자적인 데이터이다.
- 라. 존재 목적이나 유용성 면에서 필수적인 데이터이다.

6. 데이터베이스의 정의와 관계없는 것은?

- 가. 데이터베이스는 통합된 데이터이다.
- 나. 데이터베이스는 공용 데이터이다.
- 다. 데이터베이스는 운영 데이터이다.
- 라. 데이터베이스는 실시간 처리 데이터이다.

## [DB 1강-데이터베이스 정의]

7. 데이터웨어하우스(Datawarehouse)의 설명으로 가장 적절한 것은?

- 가. 제품의 생산을 위한 프로세스를 전산화해서 부품 조달에서 생산계획, 납품, 재고관리 등을 효율적으로 처리할 수 있는 공급망 관리 솔루션
- 나. 데이터 수집이나 보고를 위해 작성된 각종 양식, 보고서 관리, 문서보관 등 여러 형태의 문서관리를 수행
- 다. 대량의 데이터로부터 각종 기법, 등을 이용하여 숨겨져 있는 데이터간의 상호 관련성, 패턴, 경향 등의 유용한 정보를 추출하여 의사결정에 적용
- 라. 기간 업무 시스템에서 추출되어 새로이 생성된 데이터베이스로서 의사결정자원 시스템을 지원하는 주체적, 통합적, 시간적 데이터의 집합체

8. 데이터베이스의 등장 이유로 보기 어려운 것은?

- 가. 여러 사용자가 데이터를 공유해야 할 필요가 생겼다.
- 나. 데이터의 수시적인 구조 변경에 대해 응용 프로그램을 매번 수정하는 번거로움을 줄여 보고 싶었다.
- 다. 데이터의 가용성 증가를 위해 중복을 허용하고 싶었다.
- 라. 물리적인 주소가 아닌 데이터 값에 의한 검색을 수행하고 싶었다.

[정답] 7.라 8.다

## [DB 1강-데이터베이스 정의]

[기-08년4회][기-08년2월]

9. 다음 영어 설명 중 데이터베이스의 정의로 옳은 내용을 모두 나열한 것은?

- ① Integrated Data      ② Stored Data  
③ Operational Data    ④ Shared Data

가. ①,②,④

나. ②,③

다. ①,③

라. ①,②,③,④

[산-08년4회]

11. 데이터베이스의 정의 중 다음 설명에 해당하는 것은?

“데이터베이스는 단순한 데이터의 집합이 아니라 그 조직의 존재 목적이나 기능을 수행하는데 없어서는 안 될 데이터의 집합이다.”

가. 공용 데이터

나. 통합 데이터

다. 저장 데이터

라. 운영 데이터

[산-08년1월]

10. 데이터베이스는 어느 한 조직의 여러 응용 시스템들이 공유할 수 있도록 통합되고, 저장된 운영 데이터의 집합이라고 정의할 수 있다. 이 정의가 함축하고 있는 의미 중 효율성 증진을 위하여 불가피하게 최소의 중복(Minimal Redundancy) 또는 통제된 중복(Controlled Redundancy)을 허용하는 것으로 설명되는 항목은?

가. 저장된 데이터(Stored Data)

나. 공유되는 데이터(Shared Data)

다. 통합된 데이터(Integrated Data)

라. 운영 데이터(Operational Data)

[데이터베이스 정의]

- 통합된 데이터(Integrated data) : 자료의 중복을 배제 또는 최소화 한 데이터의 집합
- 저장된 데이터(Stored data) : 컴퓨터가 접근할 수 있는 저장 매체에 저장된 데이터의 집합
- 운영 데이터(Operational data) : 조직의 업무를 수행하는데 있어서 존재 가치가 확실하고 없어서는 안 될 반드시 필요한 데이터의 집합
- 공용 데이터(Shared data) : 여러 응용 시스템들이 공동으로 소유하고 유지하는 데이터의 집합

[정답] 9.라 10.다 11.라

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

### 1. DBMS 정의 ★☆☆☆☆

[출제빈도 '상']

: 응용프로그램(사용자)와 데이터베이스 사이에서 사용자의 요구에 따라 DB생성, 관리해 주는 S/W

- 응용 프로그램과 데이터베이스 사이에서 중재자

### 2. DBMS 필수 기능 ★

1) 정의 : 데이터베이스 자료형, 데이터 구조, 이용방법, 제약조건을 명시

- 다양한 응용 프로그램과 데이터베이스가 서로 인터페이스를 할 수 있는 방법을 제공

- 데이터의 논리적 구조와 물리적 구조 사이의 변환이 가능하도록 두 구조 사이의 사상(Mapping)을 명세

2) 조작 : 검색, 저장, 삭제, 갱신 기능

3) 제어 : 데이터의 무결성, 보안, 정확성, 병행수행, 안정성 유지

### 3. DBMS(DB) 장점 ★☆☆☆☆

1) 독립성 보장(논리적 or 물리적)

2) 데이터 중복이 없어짐

3) 공동으로 자료 이용

4) 일관성을 유지

5) 데이터의 무결성을 유지

6) 데이터를 표준화시킬 수 있음

7) 데이터를 통합하여 관리

8) 최신의 데이터 유지

9) 데이터 실시간 처리

### 4. DBMS(DB) 단점 ★☆☆☆☆

1) 전문가가 부족

2) 전산화 비용이 증가

3) Access할 때 오버헤드 발생

4) 시스템이 복잡

→ 예비와 회복 기법이 어려움, 자료처리 복잡

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

### 1. 스키마 정의 ★☆☆☆☆

: 데이터베이스의 구조와 제약조건에 대한 명세를 기술한 것 (설계도)

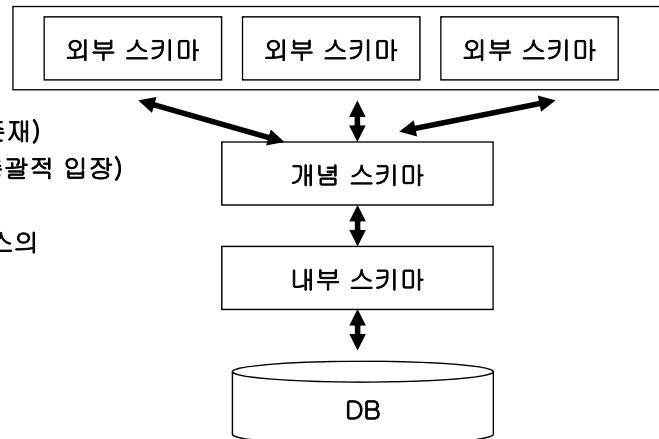
- 데이터베이스를 구성하는 데이터 객체, 이들의 성질, 이들 간에 존재하는 관계, 그리고 데이터의 조작 또는 이들 데이터 값들이 갖는 제약조건에 관한 정의를 총칭하는 용어

### 2. 스키마 특징 ★☆☆☆☆

- 1) 데이터 사전(= 시스템 카탈로그)에 저장
  - 데이터베이스에 저장되어 있는 모든 데이터 개체들에 대한 정보를 유지, 관리하는 시스템
- 2) 데이터베이스의 구조(개체, 속성, 관계)에 대한 정의
- 3) 다른 이름으로 메타데이터(데이터의 데이터)라고 함

### 3. 스키마 3계층 ★

- 1) **외부 스키마** (= 서브 스키마 = 사용자 뷰)
  - **사용자**가 보는 관점 (사용자에 따라 다름, 여러 개 존재)
- 2) **개념 스키마** (= 스키마 = 전체적인 뷰, 범기관적, 총괄적 입장)
  - DB **전체적인 논리적** 구조
  - 개체간의 관계와 제약 조건을 나타내고 데이터베이스의 접근 권한, 보안 및 무결성 규칙을 명세화 한다
- 3) **내부 스키마** (실제 Data를 저장)
  - DB 전체적인 **물리적** 구조
  - DBA 관리



## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

### 1. 데이터 언어 (= 데이터베이스 언어)

: DBMS의 3가지 기능을 구현하기 위한 언어

### 2. 데이터 언어 종류 ★★★★★

- 1) 데이터 정의어 (DDL Data Definition Language) - DBA
  - : 데이터의 형태, 구조, 데이터베이스의 저장에 관한 내용 정의 및 변경
  - **사용자**(응용 프로그램)과 **DB** 간의 인터페이스 제공
- 2) 데이터 조작어 (DML Data Manipulation Language) - 응용프로그래머, 사용자
  - : 사용자의 요구에 따라 검색, 갱신, 삽입, 삭제 등을 지원하는 기능
  - **사용자**(응용 프로그램)과 **DBMS** 간의 인터페이스 제공
- 3) 데이터 제어어 (DCL Data Control Language) - DBA
  - : 정확성과 안정성을 유지하는 기능.(무결성 유지, 보안, 권한, 병행 수행 제어, 회복)

\* 질의어(Query) : 데이터 언어 중에서 터미널에서 주로 이용하는 비절차적 데이터 조작어

### 3. 데이터베이스 사용자 ★☆☆☆☆

- 1) 응용 프로그래머 : DB 활용, 사용자 인터페이스 제공
- 2) 일반 사용자
- 3) DB 관리자 (DBA : DataBase Administrator)
  - 데이터베이스 설계와 조작에 대한 책임
  - 행정적 책임
  - 시스템 감시 및 성능 분석 (**사용자 감시 X**)

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]



### 1. DBA의 역할

: 정의 : 데이터베이스 시스템의 모든 관리와 운영에 대한 책임을 지고 있는 사람이나 그룹으로서 DDL, DCL 을 사용

- 데이터베이스 스키마 정의
- 보안 정책과 무결성(integrity) 유지
- 예비조치(backup)와 회복(recovery)에 대한 절차수립
- 데이터베이스 설계와 운영
- 사용자의 요구와 불평을 청취하고 해결
- 시스템 감시 및 성능 분석 (사용자 요구 변화 분석, 장비성능 감시, 데이터 사용 추세 분석)
- DBMS 관리
- 데이터베이스 구조 관리
- 데이터베이스 데이터 사전 구성
- 저장 구조와 액세스 방법 정의
- 데이터베이스의 이상 현상 감시
- 응용프로그램 개발(X)
- 주로 데이터 조작용(DML)을 이용(X)
- 데이터베이스 자원 활용(사용) 및 사용자의 인터페이스 제공(X)
- 데이터를 저장하고 저장된 데이터를 사용(X)
- 사용자 통제 및 감시(X)
- 정보추출을 위한 데이터베이스 접근(X)

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

1. 데이터베이스 관리 시스템의 필수 기능 중 다양한 응용 프로그램과 데이터베이스가 서로 인터페이스를 할 수 있는 방법을 제공하는 기능은?

- 가. 정의 기능      나. 조작 기능
- 다. 제어 기능      라. 저장 기능

2. 데이터베이스 관리 시스템(DBMS)의 주요 필수기능과 거리가 먼 것은?

- 가. 데이터베이스 구조를 정의할 수 있는 정의 기능
- 나. 데이터 사용자의 통제 및 보안 기능
- 다. 데이터베이스 내용의 정확성과 안정성을 유지할 수 있는 제어 기능
- 라. 데이터 조작용으로 데이터베이스를 조작할 수 있는 조작 기능

3. 데이터베이스 구성의 장점이 아닌 것은?

- 가. 데이터 중복 최소화
- 나. 여러 사용자에 의한 데이터 공유
- 다. 데이터간의 종속성 유지
- 라. 데이터 내용의 일관성 유지

4. 데이터베이스 관리 시스템(DBMS)의 필수 기능 중 제어기능에 대한 설명으로 거리가 먼 것은?

- 가. 데이터베이스를 접근하는 갱신, 삽입, 삭제 작업이 정확하게 수행되어 데이터의 무결성이 유지되도록 제어해야 한다.
- 나. 데이터의 논리적 구조와 물리적 구조 사이에 변환이 가능하도록, 두 구조 사이의 사상(Mapping)을 명시하여야 한다.
- 다. 정당한 사용자가 허가된 데이터만 접근할 수 있도록 보안(Security)을 유지하고 권한(Authority)을 검사할 수 있어야 한다.
- 라. 여러 사용자가 데이터베이스를 동시에 접근하여 데이터를 처리할 때 처리 결과가 항상 정확성을 유지하도록 병행 제어(Concurrency Control)를 할 수 있어야 한다.

5. DBMS를 사용했을 때의 장점으로 거리가 먼 것은?

- 가. 표준화의 범기관적 시행
- 나. 단순한 예비와 회복 기법
- 다. 데이터의 보안 보장이 용이
- 라. 데이터 무결성 및 일관성 유지

[정답] 1.가 2.나 3.다 4.나 5.나

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

6. 스키마(Schema)에 대한 설명으로 옳지 않은 것은?

- 가. 데이터베이스를 운용하는 소프트웨어이다.
- 나. 데이터 사전(Data Dictionary)에 저장된다.
- 다. 다른 이름으로 메타데이터(Meta-Data)라고도 한다.
- 라. 데이터베이스의 구조(개체, 속성, 관계)에 대한 정의이다.

7. 데이터베이스 내용에 대한 전체적인 뷰(View)라고 볼 수 있는 스키마는?

- 가. 외부 스키마      나. 개념 스키마
- 다. 내부 스키마      라. 서브 스키마

8. 데이터베이스 구조를 전반적으로 기술한 것을 스키마라고 한다. 3층 스키마에 해당하지 않는 것은?

- 가. 외부 스키마      나. 개념 스키마
- 다. 논리 스키마      라. 내부 스키마

9. 개체간의 관계와 제약 조건을 나타내고 데이터베이스의 접근 권한, 보안 및 무결성 규칙 명세가 있는 스키마는?

- 가. 내부 스키마      나. 외부 스키마
- 다. 개념 스키마      라. 서브 스키마

10. 스키마(Schema)에 대한 설명으로 옳지 않은 것은?

- 가. 스키마(schema)-데이터베이스의 구조와 제약 조건에 대한 명세(specification)를 기술한 것이다.
- 나. 외부스키마(external schema)-전체 데이터베이스의 한 논리적인 부분으로 볼 수 있으므로 서브스키마(subschema)라고도 한다.
- 다. 내부 스키마(internal schema)-사용자나 응용 프로그램이 접근할 수 있는 정의를 기술한다.
- 라. 개념 스키마(conceptual schema)-데이터베이스 접근 권한, 보안정책, 무결성 규칙을 명세화 한다.

11. 개념 스키마(conceptual schema)에 대한 설명으로 옳지 않은 것은?

- 가. 단순 스키마(Schema)라고도 한다.
- 나. 범기관적 입장에서 데이터베이스를 정의한 것이다.
- 다. 모든 응용 시스템과 사용자가 필요로 하는 데이터를 통합한 조직 전체의 데이터베이스로 하나만 존재한다.
- 라. 개개 사용자나 응용 프로그램이 접근하는 데이터베이스를 정의한 것이다.

[정답] 6.가 7.나 8.다 9.다 10.다 11.라

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

12. 사용자나 응용 프로그램이 각 개인의 입장에서 필요로 하는 데이터베이스의 논리적 구조를 나타내는 것은?

- 가. 외부 스키마      나. 개념 스키마
- 다. 내부 스키마      라. 처리 스키마

13. 데이터베이스 시스템의 3단계 구조인 내부 스키마(Internal Schema), 개념 스키마(Conceptual Schema), 외부 스키마(External Schema)에 대한 설명의 연결이 옳은 것은?

- ㉠ 내부 스키마    ㉡ 개념 스키마    ㉢ 외부 스키마
- ① 데이터의 실제 저장 방법을 기술
- ② 저장된 데이터와 그들 간의 관계를 기술
- ③ 데이터베이스 사용자의 관점을 기술

가. ㉠-②    나. ㉡-①    다. ㉢-③    라. ㉠-③

14. 데이터베이스를 정의하는 과정에서 주로 사용되는 데이터 언어는?

- 가. DDL      나. DCL      다. DML      라. DQL

[정답] 12.가 13.다 14.가 15.나 16.라 17.라

15. 사용자로 하여금 데이터를 처리할 수 있게 하는 도구로서 사용자(응용프로그램)와 DBMS간의 인터페이스를 제공하는 언어는?

- 가. 데이터 정의어(DDL)      나. 데이터 조작어(DML)
- 다. 데이터 부속어(DSL)      라. 데이터 제어어(DCL)

16. 데이터 제어어(DCL)의 역할이 아닌 것은?

- 가. 불법적인 사용자로부터 데이터를 보호하기 위한 데이터 보안(Security)
- 나. 데이터 정확성을 위한 무결성(Integrity)
- 다. 시스템 장애에 대비한 데이터 회복과 병행 수행
- 라. 데이터의 검색, 삽입, 삭제, 변경

17. 데이터베이스 관리 시스템에서 데이터 언어(Data Language)에 대한 설명으로 옳지 않은 것은?

- 가. 데이터 정의어(DDL)는 데이터베이스를 정의하거나 그 정의를 수정할 목적으로 사용하는 언어이다.
- 나. 데이터베이스를 정의하고 접근하기 위해서 시스템과의 통신 수단이 데이터 언어이다.
- 다. 데이터 조작어(DML)는 사용자와 데이터베이스 관리 시스템간의 인터페이스를 제공한다.
- 라. 데이터 제어어(DCL)는 주로 응용 프로그램과 일반 사용자가 사용하는 언어이다.



## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

18. 데이터베이스 관리자의 역할로 거리가 먼 것은?

- 가. 개념 및 내부 스키마 정의
- 나. 변화 요구에 대한 적응과 성능 향상에 대한 감시
- 다. 백업 및 회복 전략의 정의
- 라. 데이터베이스 자원 활용 및 사용자의 인터페이스 제공

19. 데이터베이스 관리자(Database Administrator)의 역할로 거리가 먼 것은?

- 가. 사용자의 요구 및 불평 해결
- 나. 데이터베이스의 이상 현상 감시
- 다. 장애 시 회복에 대한 전략 수립
- 라. 응용 프로그램 구현

20. DBA(Data Base Administrator)의 기능으로 거리가 먼 것은?

- 가. 데이터베이스의 스키마를 정의
- 나. 사용자 통제 및 감시
- 다. 보안 검사와 무결성 검사 정의
- 라. 예비조치(Backup)와 회복(Recovery)에 대한 전략 정의

21. DBA의 여러 업무 중 시스템 감시 및 성능 분석 업무 내용에 해당되지 않는 것은?

- 가. 사용자 요구 변화 분석
- 나. 장비 성능 감시
- 다. 백업/회복 절차 이행
- 라. 데이터 사용 추세 분석

22. DBMS의 제어 기능에 대한 설명으로 잘못된 것은?

- 가. 모든 사용자 누구나가 접근할 수 있도록 데이터를 관리한다.
- 나. 데이터의 무결성이 파괴되지 않도록 제어한다.
- 다. 데이터의 내용에 대한 정확성과 안전성을 유지할 수 있도록 제어한다.
- 라. 여러 사용자가 데이터베이스를 동시에 접근하여 데이터를 처리하기 위한 병행제어를 한다.

[정답] 18.라 19.라 20.나 21.다 22.가

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

23. DBA의 역할이 아닌 것은?

- 가. 자료의 보안성, 무결성 유지
- 나. 스키마의 정의
- 다. 응용프로그램의 설계 및 개발
- 라. 데이터 사전의 유지 및 관리

24. 데이터베이스 관리자(DBA)의 역할에 관한 설명 중 옳지 않은 것은?

- 가. 주로 데이터 조작어(DML)를 이용하여 데이터베이스의 무결성을 유지한다.
- 나. 데이터베이스 설계와 운영을 한다.
- 다. 사용자의 요구와 불평을 청취하고 해결한다.
- 라. 시스템 감시 및 성능분석을 한다.

25. 데이터베이스 관리자(DBA)의 임무로 거리가 먼 것은?

- 가. 개념 스키마 및 내부 스키마를 정의한다.
- 나. 데이터를 저장하고 저장된 데이터를 사용한다.
- 다. 장애에 대비한 예비조치와 회복에 대한 전략을 수립한다.
- 라. 접근 권한을 부여한다.

[정답] 23.다 24.가 25.나

## [DB 2강-DBMS, 스키마, 데이터언어, DBA]

[산-08년9월]

26. 데이터베이스 관리자(DBA)의 업무로 거리가 먼 것은?

- 가. 스키마를 정의한다.
- 나. 저장구조와 액세스 방법을 선정한다.
- 다. 데이터의 단순 검색, 데이터의 삽입, 삭제, 갱신 작업을 수행한다.
- 라. 보안 및 권한 부여 정책, 데이터의 유효성 검사 방법을 수립한다.

[기-08년5월]

27. DBMS의 필수 기능에 해당하지 않는 것은?

- 가. Definition facility      나. Relation facility
- 다. Control facility        라. Manipulation facility

[산-08년5월]

28. 데이터베이스 스키마의 설명으로 옳지 않은 것은?

- 가. 스키마는 데이터베이스의 구조와 제약조건에 관한 전반적인 명세를 기술한다.
- 나. 외부 스키마는 응용프로그래머가 데이터베이스를 바라보는 관점이다.
- 다. 개념 스키마는 조직이나 기관의 총괄적 입장에서 본 데이터베이스의 전체적인 논리적 구조이다.
- 라. 하나의 데이터베이스 시스템에는 내부, 외부, 개념 스키마가 각각 하나씩만 존재한다.

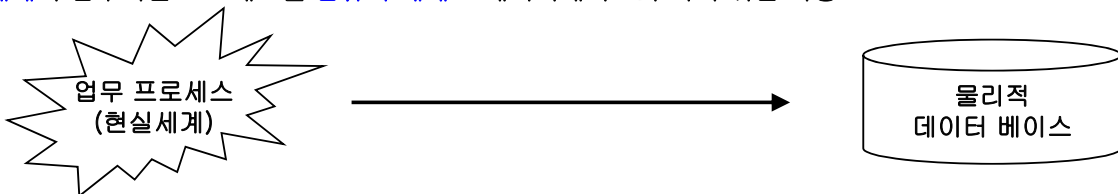
[정답] 26.다 27.나 28.라

## [DB 3강-DB설계, 데이터모델, ER모델]

[출제빈도 '상']

### 1. 데이터베이스 설계

현실 세계의 업무적인 프로세스를 컴퓨터 세계로 데이터베이스화 하기 위한 과정



### 2. 데이터베이스 설계 순서



요구조건 분석 → 개념적 설계 → 논리적 설계 → 물리적 설계 → 구현 → 운영 → 감시 및 개선

### 3. 요구조건 분석

업무 프로세스 분석 → 요구조건 명세서 작성

[요구조건 명세서] 한국대학교의 주된 개체는 학생과 과목이다. 학생은 고유의 학번이 부여되며, 추가로 이름, 전화번호 정보를 가진다. 과목은 고유의 과목코드가 부여되며, 추가로 과목명, 과목 내용을 가진다. 한 명의 학생은 여러 개의 과목을 수강할 수 있다.

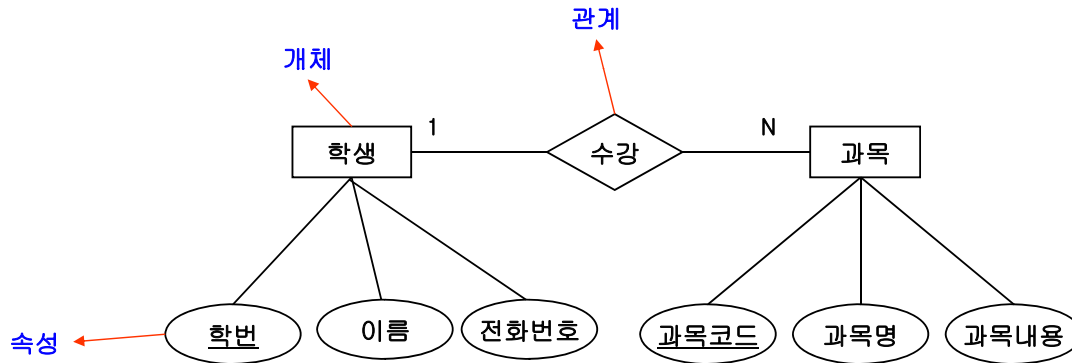
## [DB 3강-DB설계, 데이터모델, ER모델]

### 4. 개념적 설계 ★★☆☆☆

: 개체 타입과 이들 간의 관계 타입을 이용해 현실 세계를 개념적으로 표현 (산출물 : 개체관계도 = ER 다이어그램)

- DBMS에 독립적인 개념 스키마 모델링

- **트랜잭션 모델링**



(1) **개체 (Entity)** : 데이터베이스에 표현하려고 하는 현실 세계의 대상체

- 예) 학생, 교수, 학과, 과목

(2) **속성 (Attribute)** : 개체(Entity)의 성질, 분류, 식별, 수량, 상태 등을 나타내는

- 예) 학생 - 학번, 이름, 전화번호

(3) **관계 (Relationship)** : 두 개체 간에 의미 있는 연결

- 예) 학생은 과목을 수강한다. 과목은 학생에게 수강되어 진다.

## [DB 3강-DB설계, 데이터모델, ER모델]

### 5. 논리적 설계 ★★☆☆☆

: 목표 DBMS에 맞추어 논리적 모델로 설계 (관계형, 계층형, 망형 모델)

- **트랜잭션 인터페이스 설계**

- **스키마**의 평가 및 정제

- 정규화 과정 수행

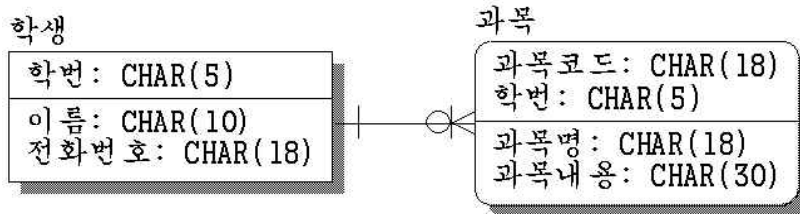


## [DB 3강-DB설계, 데이터모델, ER모델]

### 6. 물리적 설계 ★★★★★

: 저장레코드 양식의 설계 및 물리적 구조 데이터 표현

- 설계 시 고려사항 : 응답시간, 저장공간의 효율성, 트랜잭션의 처리량
- 어떤 인덱스를 만들 것인지에 대한 고려
- 성능 향상을 위한 개념 스키마의 변경 여부 검토
- 빈번한 질의와 트랜잭션들의 수행 속도를 높이기 위한 고려
- 접근 경로 설계
- 레코드 집중의 분석 및 설계
- **트랜잭션 세부사항 설계**



### 7. 구현 ★★★★★

: 목표 DBMS DDL로 스키마를 작성(정의), 응용 프로그램을 위한 트랜잭션을 작성하는 단계

## [DB 3강-DB설계, 데이터모델, ER모델]

### 1. 데이터 모델 정의 ★★★★★

: 현실 세계의 데이터 구조를 컴퓨터 세계의 데이터 구조로 기술하는 개념적인 도구이다.

### 2. 데이터 모델 종류 ★★★★★

- 1) **개념적 모델** : 현실세계를 추상적으로 표현 (E-R 모델)
- 2) **논리적 모델** : 개념적 모델을 컴퓨터가 이해할 수 있도록 표현
  - 관계모델, 계층모델, 네트워크(망) 모델

### 3. 데이터 모델 구성요소 ★★★★★

- 1) **구조 (Structure)** : 개체들 간의 관계
- 2) **연산 (Operation)** : 데이터 처리하는 방법
- 3) **제약조건 (Constraint)** : 실제 데이터의 논리적인 제약조건

속성 (Attribute) = 항목, 필드

### 4. 데이터 모델 용어



## [DB 3강-DB설계, 데이터모델, ER모델]

### 1. ER (Entity-Relationship, 개체관계도) 모델

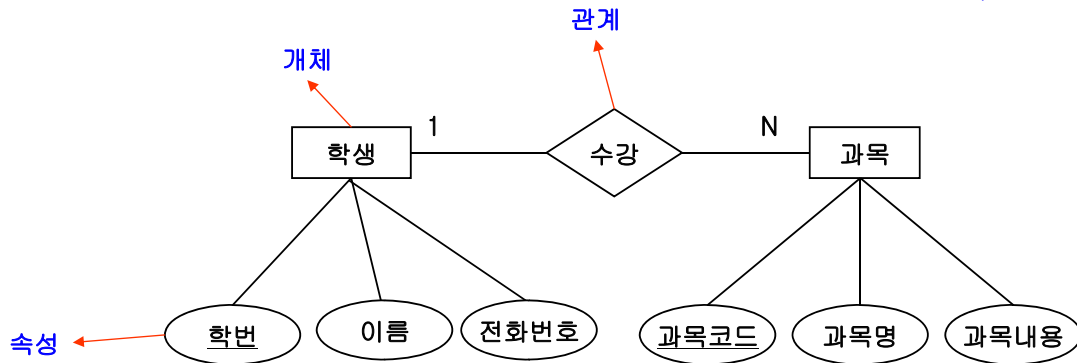


: 개체와 개체간의 관계를 도식화한다.

- 1976년 P.Chen에 의해 처음으로 제안

| 의 미 | 개체  | 관계  | 속성    | 기본키 속성 | 연결, 링크 |
|-----|-----|-----|-------|--------|--------|
| 기 호 | 사각형 | 마름모 | 타원, 원 |        | 선      |

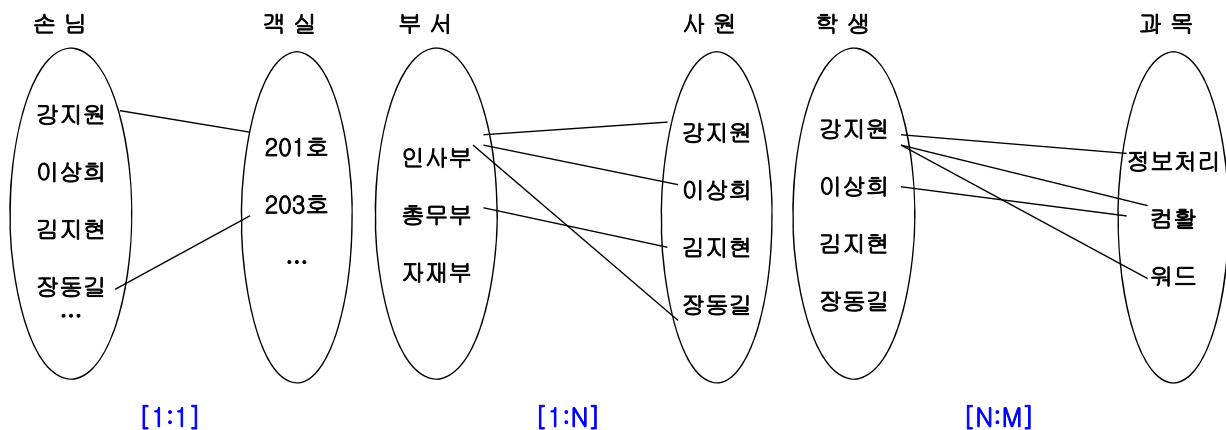
기본키 : 고유의 속성(ex. 아이디, 학번)



## [DB 3강-DB설계, 데이터모델, ER모델]

### 2. 관계의 종류 (관계 타입) ★☆☆☆☆

- 1) **1:1 관계** : E1에 있는 한 개의 데이터는 E2에 있는 한 개의 데이터와 일치하는 관계이다.
- 2) **1:N 관계** : E1에 있는 각각의 데이터는 E2에 있는 하나 이상의 데이터와 일치하나 E2에 있는 데이터는 E1에 있는 데이터와 단지 하나 만이 일치하는 관계이다.
- 3) **N:M 관계** : E1에 있는 각각의 데이터는 E2에 있는 하나 이상의 데이터와 일치하고 E2에 있는 데이터도 E1에 있는 하나 이상의 데이터와 일치하는 관계이다.



## [DB 3강-DB설계, 데이터모델, ER모델]

1. 데이터 모델(Data Model)의 개념으로 가장 적절한 것은?

- 가. 현실 세계의 데이터 구조를 컴퓨터 세계의 데이터 구조로 기술하는 개념적인 도구이다.
- 나. 컴퓨터 세계의 데이터 구조를 현실 세계의 데이터 구조로 기술하는 개념적인 도구이다.
- 다. 현실 세계의 특정한 한 부분의 표현이다.
- 라. 가상 세계의 데이터 구조를 현실 세계의 데이터 구조로 기술하는 개념적인 도구이다.

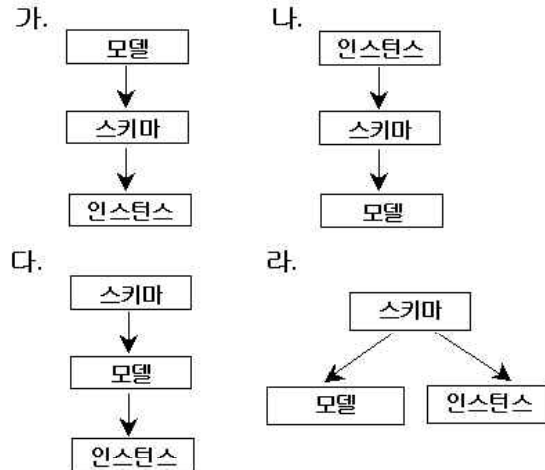
2. 현실 세계에 존재하는 개체를 인간이 이해 할 수 있는 정보 구조로 표현하는 과정을 무엇이라 하는가?

- 가. 데이터 모델링(Data Modeling)
- 나. 정보 모델링(Information Modeling)
- 다. 데이터 구조화(Data Structuring)
- 라. 정보 구조화(Information Structuring)

3. 데이터 모델의 구성 요소가 아닌 것은?

- 가. 논리적으로 표현된 데이터 구조
- 나. 구성 요소의 연산
- 다. 구성 요소의 제약 조건
- 라. 물리적 저장 구조

4. 데이터 모델, 스키마, 인스턴스 간의 관계로 옳은 것은?



5. 데이터의 가장 작은 논리적 단위로서 파일 구조상의 데이터 항목 또는 데이터 필드에 해당하는 것은?

- 가. tuple      나. relation
- 다. domain    라. attribute

[정답] 1.가 2.나 3.라 4.가 5.라

## [DB 3강-DB설계, 데이터모델, ER모델]

6. 데이터베이스의 구성 요소 중 개체(Entity)에 대한 설명으로 적합하지 않은 것은?

- 가. 속성들이 가질 수 있는 모든 값들의 집합이다.
- 나. 데이터베이스에 표현하려고 하는 현실 세계의 대상체이다.
- 다. 유형, 무형의 정보로서 서로 연관된 몇 개의 속성으로 구성된다.
- 라. 파일의 레코드에 대응하는 것으로 어떤 정보를 제공하는 역할을 수행한다.

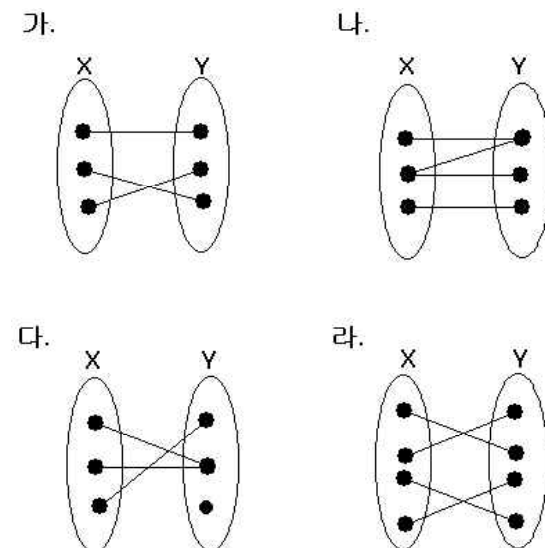
7. 개체-관계(E-R) 모델에 대한 설명으로 잘못 된 것은?

- 가. E-R 다이어그램으로 표현하며 P.Chen이 제안했다.
- 나. 일 대 일(1:1) 관계 유형만을 표현 할 수 있다.
- 다. 개체 타입과 이들 간의 관계 타입을 이용해 현실 세계를 개념적으로 표현한 방법이다.
- 라. E-R 다이어그램은 E-R 모델을 그래프 방식으로 표현한 것이다.

8. 개체-관계 모델(E-R Model)에 대한 설명으로 옳지 않은 것은?

- 가. 개체와 개체간의 관계를 도식화한다.
- 나. 개체 집합을 사각형으로 표시한다.
- 다. 관계를 다이아몬드로 표시한다.
- 라. 일 대 일(1:1), 일 대 다(1:N) 관계 유형만 표현할 수 있다.

9. 다음 두 집합 X와 Y의 대응관계를 보인 그림 중 다 대 다(N:M)의 관계는?



[정답] 6.가 7.나 8.라 9.나

## [DB 3강-DB설계, 데이터모델, ER모델]

10. E-R 모델에 대한 설명으로 옳지 않은 것은?

- 가. 개체 타입과 이들 간의 관계 타입을 이용해서 현실 세계를 개념적으로 표현하는 방법이다.
- 나. 관계 타입을 표현하는 방법은 그 관계 타입의 이름과 함께 연관된 개체 타입들을 링크로 연결한다.
- 다. 관계 타입의 차원은 관계 타입과 관련된 엔티티 타입의 개수이다.
- 라. 관계 인스턴스는 다른 엔티티 타입에 속한 엔티티 사이의 관계를 표현한다.

11. E-R 다이어그램의 구성 요소에 대한 표현이 틀리게 짝지어진 것은?

- 가. 개체 집합-직사각형
- 나. 관계 집합-마름모꼴
- 다. 속성-원
- 라. 링크-화살표

12. 학생과 학교 개체간의 학적 관계를 E-R 다이어그램으로 옳게 표현한 것은?

- 가.
- 나.
- 다.
- 라.

13. 다음은 어느 신용카드 회사에서 고객과 고객이 발급받은 신용카드간의 관계를 나타내는 E-R 다이어그램이다. 이에 대한 설명 중 부적합한 것은?



- 가. 신용카드를 발급받지 않은 고객은 존재할 수 없다.
- 나. 고객이 없는 신용카드는 발급될 수 없다.
- 다. 개체 고객과 신용카드간의 카드발급 관계는 일 대 일 관계이다.
- 라. 한 사람의 고객에는 반드시 하나의 신용카드만 발급된다.

14. 데이터 모델에 표시할 요소로 가장 타당한 것은?

- 가. 개체, 속성, 관계
- 나. 정의, 조작, 제어
- 다. 구조, 연산, 제약조건
- 라. 개체, 관계, 구조

[정답] 10.나 11.라 12.가 13.가 14.다

## [DB 3강-DB설계, 데이터모델, ER모델]

15. 데이터베이스 설계의 논리적 설계 단계에서 수행하는 작업이 아닌 것은?

- 가. 논리적 데이터 모델로 변환
- 나. 트랜잭션 인터페이스 설계
- 다. 스키마의 평가 및 정제
- 라. 트랜잭션 모델링

16. 현실 세계의 객체를 개념적으로 표현할 때 기본적으로 개체타입과 이들 간의 관계를 이용하도록 P. Chen 이 제안한 데이터 모델은?

- 가. 개체-관계 모델
- 나. 계층형 데이터 모델
- 다. 관계형 데이터 모델
- 라. 네트워크형 데이터 모델

17. 하위 수준(물리적)의 데이터 모델에 관한 설명으로 가장 올바른 것은?

- 가. 데이터에 대한 추상적인 개념을 제공한다.
- 나. 엔티티, 속성, 관계 등의 개념을 사용한다.
- 다. 레코드의 형식, 순서, 접근 경로와 같은 정보를 사용하여 데이터가 컴퓨터에 저장되는 방법을 묘사한다.
- 라. 데이터를 테이블 형태로 표현한다.

18. 데이터베이스 설계 단계 중 물리적 설계 단계에 해당하지 않는 것은?

- 가. 저장 레코드 양식 설계
- 나. 접근 경로 설계
- 다. 레코드 집중의 분석 및 설계
- 라. 트랜잭션 인터페이스 설계

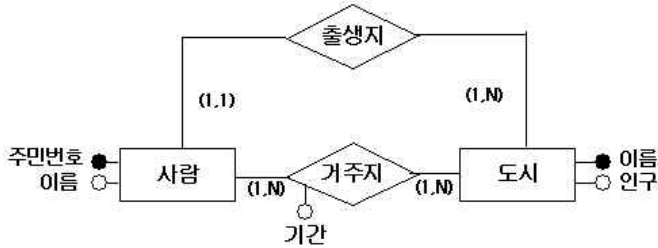
19. 물리적 데이터베이스 구조에 대한 설명으로 잘못된 것은?

- 가. 기본적인 데이터 단위는 저장 파일이다.
- 나. 데이터베이스 시스템의 성능에 중대한 영향을 미친다.
- 다. 여러 가지 타입의 저장 레코드 집합이라는 면에서 단순한 파일과 다르다.
- 라. 데이터베이스에 포함될 여러 파일 타입에 대한 저장 레코드의 양식, 순서, 접근 경로를 표현한 것이다.

[정답] 15.라 16.가 17.다 18.라 19.가

## [DB 3강-DB설계, 데이터모델, ER모델]

20. 다음 E-R 다이어그램에 대한 설명 중 거리가 먼 것은?



- 가. 개체 사람의 기본키는 주민번호이다.
- 나. 사람과 도시 사이의 거주지 관계는 다 대 다 관계이다.
- 다. 거주지 관계를 관계 데이터 모델로 표현할 때 별도의 릴레이션으로 모델링하는 것이 일반적이다.
- 라. 출생지 관계를 관계 데이터 모델로 표현할 때 별도의 릴레이션으로 모델링한다면 기본키는 주민번호와 도시이름이 되어야 한다.

21. 데이터베이스 설계 순서로 옳은 것은?

- 가. 요구 조건 분석→개념적 설계→논리적 설계→물리적 설계→구현
- 나. 요구 조건 분석→논리적 설계→개념적 설계→물리적 설계→구현
- 다. 요구 조건 분석→논리적 설계→물리적 설계→개념적 설계→구현
- 라. 요구 조건 분석→개념적 설계→물리적 설계→논리적 설계→구현

22. 데이터 모델 중 성격이 다른 것은?

- 가. 개체-관계(E-R)모델      나. 관계모델
- 다. 계층모델                      라. 네트워크 모델

23. 데이터베이스 설계에 있어 개념 스키마 모델링과 트랜잭션 모델링을 병행적으로 수행하는 단계는?

- 가. 요구분석 설계
- 나. 개념적 설계
- 다. 논리적 설계
- 라. 물리적 설계

24. 데이터베이스 설계 단계 중 논리적 설계 단계에 해당하는 것은?

- 가. 데이터 및 처리 요구 조건을 설계한다.
- 나. 트랜잭션을 모델링한다.
- 다. 목표 DBMS에 맞는 스키마를 설계한다.
- 라. 트랜잭션의 세부 설계를 한다.

[정답] 20.라 21.가 22.가 23.나 24.다

## [DB 3강-DB설계, 데이터모델, ER모델]

25. 물리적 데이터베이스 설계 시 고려 사항으로 가장 거리가 먼 것은?

- 가. 레코드의 크기
- 나. 파일에 대한 트랜잭션의 갱신과 참조 성향
- 다. 수행될 질의와 트랜잭션의 예상 빈도
- 라. 인덱스의 구조

26. 데이터베이스의 물리적 설계 단계에서 수행되는 작업이 아닌 것은?

- 가. 저장레코드 양식 설계
- 나. 접근 경로 설계
- 다. 레코드 집중의 분석 및 설계
- 라. 트랜잭션 인터페이스 설계

27. 물리적 데이터베이스 설계 시 그의 성능을 측정할 수 있는 척도로 거리가 가장 먼 것은?

- 가. 응답시간
- 나. 저장 공간의 효율화
- 다. 트랜잭션 처리량
- 라. 트랜잭션의 지속성

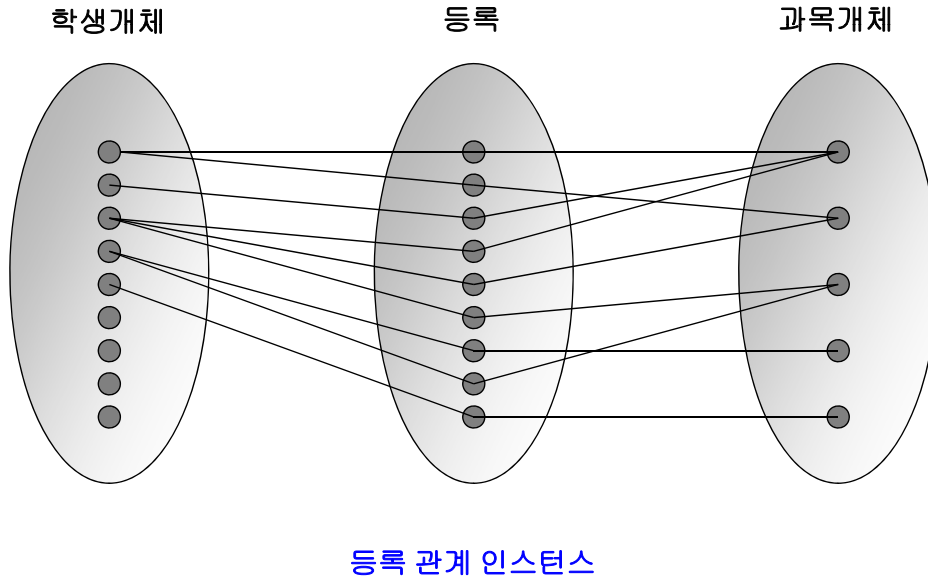
[정답] 25.다 26.라 27.라



## [DB 3강-DB설계, 데이터모델, ER모델]

E-R 모델에 대한 설명으로 옳지 않은 것은?

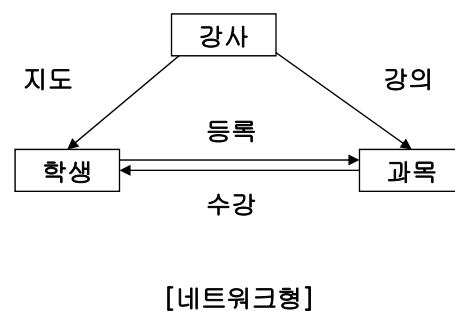
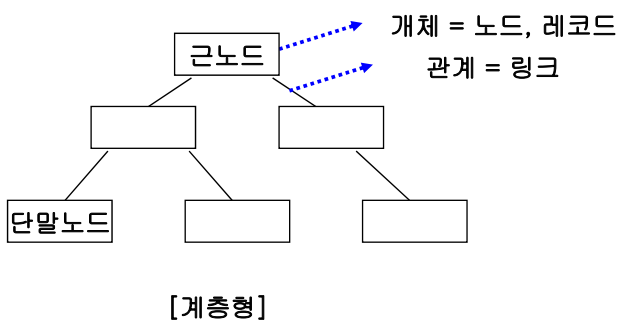
- 가. **개체 타입**과 이들 간의 **관계 타입**을 이용해서 현실 세계를 개념적으로 표현하는 방법이다.
- 나. **관계 타입을 표현하는 방법은 그 관계 타입의 이름과 함께 연관된 개체 타입들을 링크로 연결한다.**
- 다. **관계 타입의 차원은 관계 타입과 관련된 엔티티 타입의 개수이다.**
- 라. **관계 인스턴스**는 다른 **엔티티 타입**에 속한 엔티티 사이의 관계를 표현한다.



## [DB 4강-논리적데이터모델, 관계데이터모델] [출제빈도 '상']

### 1. 논리적 데이터 모델의 종류 ★★★★★

| 종 류   | 구 조                        | 관계 표현                         | 특 징                               |
|-------|----------------------------|-------------------------------|-----------------------------------|
| 관계형   | 표<br>= Table<br>= Relation | 키 (기본키, 외래키)<br>1:1, 1:N, N:M | 대표적 언어 : SQL<br>가장 많이 사용되는 데이터 모델 |
| 계층형   | 트리                         | 부모-자식 관계(족보)<br>N:M 직접 표현 불가  | 사이클이 허용되지 않음<br>개체 삭제 시 연쇄 삭제 발생  |
| 네트워크형 | 그래프, 망                     | 오너-멤버 관계<br>N:M 직접 표현 불가      | 사이클 허용<br>상/하위 개체가 복수 대응          |

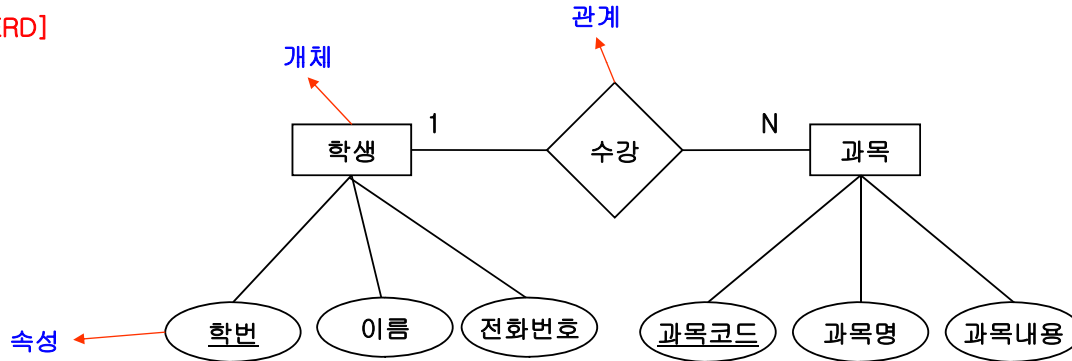


## [DB 4강-논리적데이터모델, 관계데이터모델]

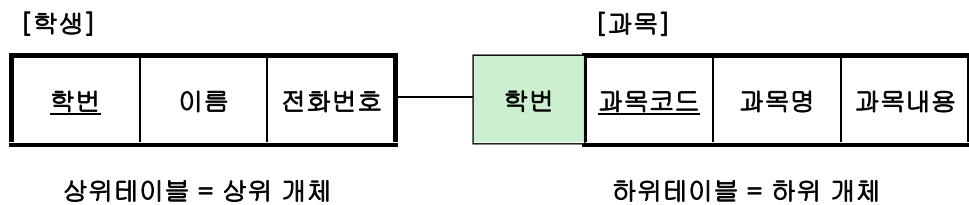
### 1. 요구분석 → 개념적 설계 → 논리적 설계

**[요구분석]** 한국학원의 주된 개체는 학생과 과목이다. 학생은 고유의 학번이 부여되며, 추가로 이름, 전화번호 정보를 가진다. 과목은 고유의 과목코드가 부여되며, 추가로 과목명, 과목내용을 가진다. 한 명의 학생은 여러 개의 과목을 수강할 수 있다.

**[ERD]**

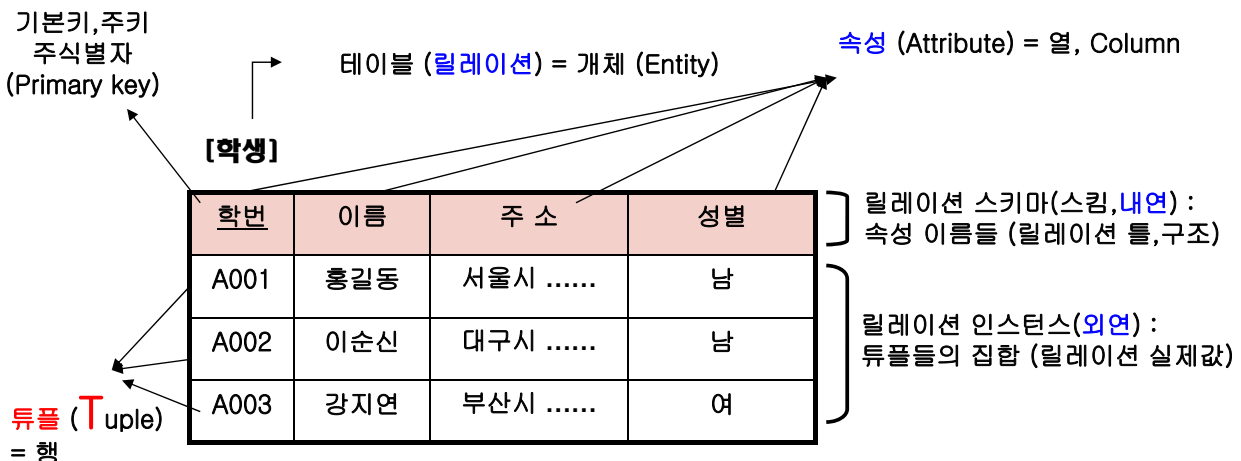


**[관계형 데이터 모델]**



## [DB 4강-논리적데이터모델, 관계데이터모델]

### 1. 관계형 데이터베이스의 릴레이션 구조



- \* 도메인 (Domain) : 한 속성에 나타날 수 있는 값들의 범위(집합)
- \* 차수 (Degree): 속성들의 수
- \* 카디널리티 (cardinality) : 튜플들의 수
- \* 널 (Null) : "해당없음" 등의 이유로 정보 부재를 나타내기 위해 사용 하는 특수한 데이터 값  
- 공백이나 0(zero) X

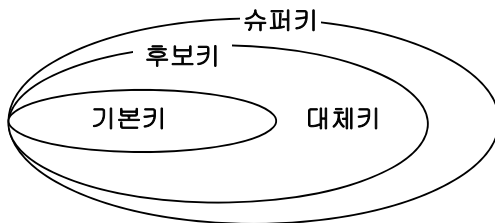
## [DB 4강-논리적데이터모델, 관계데이터모델]

### 2. 릴레이션 특징 ★★★★★

- 1) 한 릴레이션에 정의된 튜플들은 모두 다르다.
- 2) 한 릴레이션에 정의된 튜플들은 순서에 무관하다.
- 3) 튜플들은 시간에 따라 변한다.
- 4) 릴레이션 스키마를 구성하는 속성들도 순서에 무관하다.
- 5) 속성의 명칭은 유일해야 하지만, 속성의 값은 동일해도 된다.
- 6) 속성은 더 이상 쪼갤 수 없는 **원자값**으로 구성된다. (속성 값은 분해 X, 다중값 X)
- 7) 릴레이션을 구성하는 튜플을 유일하게 식별하기 위한 속성들의 부분집합을 키(Key)로 설정한다.

### 3. 키 (Key) ★★★★★☆☆

- 1) 슈퍼키 : 한 릴레이션 내에 있는 속성들의 집합으로 구성된 키를 말한다. (**유일성**)
- 2) 후보키 : 한 릴레이션 내에 있는 모든 튜플들을 유일하게 식별할 수 있는 하나 또는 몇 개의 애트리뷰트 집합  
(최소 슈퍼키 : **유일성 + 최소성**)
- 3) **기본키** : 후보키 중에 선택한 키 (중복되어서는 안되며, Null 값을 가질 수 없다.)
- 4) 대체키 : 후보키 중에서 기본키를 제외한 속성들
- 5) **외래키** : 어떤 R에서 다른 R을 참조할 때 참조 기준이 되는 속성으로서 참조하고자 하는 R의 기본키와 동일



[회원]

| 학번   | 이름  | 주 소 | 주민번호   | 전공  |
|------|-----|-----|--------|-----|
| A001 | 홍길동 | 서울시 | 123456 | 컴퓨터 |
| A002 | 이순신 | 대구시 | 222222 | 수학  |
| A003 | 강감찬 | 부산시 | 333333 | 물리  |

## [DB 4강-논리적데이터모델, 관계데이터모델]

### 4. 제약조건 (무결성) ★★★★★

성적관리 테이블

| 학번   | 이름  | 핸드폰번호        | 주소                  | 선택과목   | 담당교수 | 성적 |
|------|-----|--------------|---------------------|--------|------|----|
| K001 | 홍길동 | 011-111-1111 | 서울시 구로구 구로본동 234-24 | 자료구조   | 박세리  | A  |
| K002 | 이순신 | 016-111-1111 | 서울시 영등포구 여의도동 234-2 | 데이터베이스 | 이미연  | B  |
| K003 | 강감찬 | 018-111-1111 | 서울시 강남구 역삼동 234-2   |        |      |    |
| K004 | 선동열 | 017-111-1111 | 서울시 서초구 서초동 234-2   |        |      |    |
| K005 | 박찬호 | 019-111-1111 | 서울시 종로구 관철동 234-2   | 컴퓨터구조  | 강동원  | C  |

관계 형성을 통한 상위 테이블의 PK 전이

Primary key (기본키)

Foreign key (외래키)

학생관리 테이블

| 학번   | 이름  | 핸드폰번호        | 주소                  |
|------|-----|--------------|---------------------|
| K001 | 홍길동 | 011-111-1111 | 서울시 구로구 구로본동 234-24 |
| K002 | 이순신 | 016-111-1111 | 서울시 영등포구 여의도동 234-2 |
| K003 | 강감찬 | 018-111-1111 | 서울시 강남구 역삼동 234-2   |
| K004 | 선동열 | 017-111-1111 | 서울시 서초구 서초동 234-2   |
| K005 | 박찬호 | 019-111-1111 | 서울시 종로구 관철동 234-2   |

성적관리 테이블

| 학번   | 선택과목   | 담당교수 | 성적 |
|------|--------|------|----|
| K001 | 자료구조   | 박세리  | A  |
| K002 | 데이터베이스 | 이미연  | B  |
| K005 | 컴퓨터구조  | 강동원  | C  |

- \* **참조 무결성** : 릴레이션은 참조할 수 없는 **외래키** 값을 가질 수 없음을 의미하는 제약 조건
- \* **개체 무결성** : 한 릴레이션의 **기본키**를 구성하는 어떠한 속성 값도 널(NULL) 값이나 중복 값을 가질 수 없다
- \* **도메인 무결성** : 각 속성 값은 반드시 정의된 도메인에 속한 값이어야 한다.

## [DB 4강-논리적데이터모델, 관계데이터모델]

1. 다음의 논리적인 데이터 모델에서 데이터간의 관계를 기본키(Primary Key)와 이를 참조하는 외래키(Foreign Key)로 표현하는 데이터 모델은?

- 가. 관계형 데이터 모델
- 나. 네트워크 데이터 모델
- 다. 계층적 모델
- 라. 객체지향 데이터 모델

2. 릴레이션의 특성에 대한 설명으로 잘못된 것은?

- 가. 한 릴레이션에 포함된 튜플들은 모두 상이하다.
- 나. 한 릴레이션에 포함된 튜플 사이에는 순서가 없다.
- 다. 한 릴레이션을 구성하는 애트리뷰트 사이에는 일정한 순서가 있다.
- 라. 모든 애트리뷰트 값은 원자 값이다.

3. 개체 집합에 대한 속성 관계를 표시하기 위해 개체를 노드로 표현하고 개체 집합들 사이의 관계를 링크로 연결한 트리(Tree)형태의 자료 구조 모델은?

- 가. 망 데이터 모델
- 나. 계층 데이터 모델
- 다. 관계 데이터 모델
- 라. 객체지향 데이터 모델

4. 계층 데이터 모델에서 두 레코드 간에 직접 표현 방법을 제공하지 않는 것은?

- 가. 1:1 관계
- 나. 1:N 관계
- 다. M:N 관계
- 라. 두 개의 1:N 관계

5. 전체적인 구조가 트리 형태로 되어 있고, 두 레코드 타입 간에는 하나의 관계만 허용되는 데이터 모델은?

- 가. 관계 데이터 모델
- 나. 네트워크 데이터 모델
- 다. 계층 데이터 모델
- 라. 객체-관계 데이터 모델

[정답] 1.가 2.다 3.나 4.다 5.다

## [DB 4강-논리적데이터모델, 관계데이터모델]

6. 논리적 데이터 모델에 대한 설명으로 옳지 않은 것은?

- 가. 관계형, 계층형, 네트워크형 모델 등이 있다.
- 나. 네트워크형 모델은 레코드들이 링크에 의해서 서로 연결되는 그래프 형태로 구성된다.
- 다. 관계형 모델은 릴레이션의 집합으로 표현된다.
- 라. 계층적 모델은 다 대 다(N:M) 관계의 표현이 쉽다.

7. 논리적 데이터 모델에 대한 설명으로 옳지 않은 것은?

- 가. 관계형 데이터 모델 - 데이터베이스를 테이블의 집합으로 표현한다.
- 나. 네트워크 데이터 모델 - 데이터베이스를 그래프 구조로 표현한다.
- 다. 계층적 데이터 모델 - 데이터베이스를 계층적 그래프 구조로 표현한다.
- 라. 객체지향 데이터 모델 - 데이터베이스를 객체/상속 구조로 표현한다.

8. 계층형 데이터 모델에 관한 설명으로 옳지 않은 것은?

- 가. 계층 데이터 모델이 지원하는 스키마의 논리적 구조는 트리 형태의 자료 구조가 된다.
- 나. 계층 정의 트리는 하나의 루트 레코드 타입과 다수의 종속 레코드 타입으로 구성된 순서 트리이다.
- 다. 어떤 부모-자식 관계에서도 부모 레코드가 되지 못한 레코드 타입은 계층 정의 트리의 단말 노드이다.
- 라. 레코드 타입들 간에는 사이클(Cycle)이 허용된다.

9. 데이터 모델에 관한 설명 중 옳지 않은 것은?

- 가. 관계 데이터 모델은 개체와 관계 모두가 테이블로 표현 된다.
- 나. 계층 데이터베이스는 부자 관계(Parent-Child Relationship)를 나타내는 트리 형태의 자료 구조로 표현된다.
- 다. 네트워크 데이터베이스는 오너-멤버 관계(Owner-Member Relationship)를 나타내는 트리 구조로 표현 된다.
- 라. 데이터 모델은 데이터, 데이터의 관계, 데이터의 의미 및 일관성 제약 조건 등을 기술하기 위한 개념적 도구들의 모임이다.

10. 데이터베이스의 논리적 구조 표현을 그래프 형태로 표현하며, 일 대 다(1:N) 관계에 연관된 레코드 타입들을 각 각 오너(Owner), 멤버(Member)라고 하고, 이들의 관계를 오너-멤버 관계라고도 일컫는 데이터 모델은?

- 가. 관계형 데이터 모델
- 나. 네트워크 데이터 모델
- 다. 계층적 데이터 모델
- 라. 객체지향적 데이터 모델

[정답] 6.라 7.다 8.라 9.다 10.나

## [DB 4강-논리적데이터모델, 관계데이터모델]

11. 관계 데이터 모델에서 키에 대한 설명으로 옳지 않은 것은?

- 가. 릴레이션에 있는 모든 튜플들을 유일하여 식별할 수 있는 하나 또는 몇 개의 애트리뷰트 집합을 그 릴레이션의 후보키라 한다.
- 나. 널 값을 가지더라도 모든 튜플을 구분할 수 있으면 기본 키가 된다.
- 다. 후보키가 둘 이상 되는 경우에 그 중에서 어느 하나를 선정하여 기본키라 지정하면, 나머지 후보키들은 대체키가 된다.
- 라. 유일성만 있고 최소성이 없는 애트리뷰트 집합을 슈퍼키라 한다.

12. 상위 하나의 레코드에 대하여 하위의 레코드가 복수 대응하고, 하위 하나의 레코드에 대해서 상위 레코드도 복수 대응하는 데이터베이스 구조는?

- 가. 망구조                      나. 계층구조
- 다. 관계구조                라. 결합구조

13. 한 릴레이션의 기본키를 구성하는 어떠한 속성 값도 널(NULL) 값이나 중복 값을 가질 수 없다는 것을 의미하는 것은?

- 가. 개체 무결성 제약 조건    나. 참조 무결성 제약 조건
- 다. 보안 무결성 제약 조건    라. 정보 무결성 제약 조건

14. 릴레이션은 참조할 수 없는 외래키값을 가질 수 없음을 의미하는 제약 조건은?

- 가. 개체 무결성            나. 참조 무결성
- 다. 보안 무결성            라. 정보 무결성

15. 데이터베이스에서 아직 알려지지 않거나 모르는 값으로서 '해당 없음'등의 이유로 정보 부재를 나타내기 위해 사용하는 특수한 데이터값을 무엇이라 하는가?

- 가. 원자값(atomic value)
- 나. 참조값(reference value)
- 다. 무결값(integrity value)
- 라. 널값(null value)

16. 어떤 릴레이션 R1의 기본키의 값들과 일치함을 요구하는 다른 릴레이션 R2의 한 속성을 무엇이라 하는가?

- 가. 참조 제약(referential constraint)
- 나. 외래키(foreign key)
- 다. 기본키(primary key)
- 라. 참조 무결성(referential integrity)

[정답] 11.나 12.가 13.가 14.나 15.라 16.나

## [DB 4강-논리적데이터모델, 관계데이터모델]

17. 데이터베이스에 관련된 용어의 설명으로 옳지 않은 것은?

- 가. 튜플(Tuple) - 테이블에서 열에 해당된다.
- 나. 애트리뷰트(Attribute) - 데이터의 가장 작은 논리적 단위로서 파일 구조상의 데이터 항목 또는 데이터 필드에 해당된다.
- 다. 릴레이션(Relation) - 릴레이션 스키마와 릴레이션 인스턴스로 구성된다.
- 라. 도메인(Domain) - 애트리뷰트가 취할 수 있는 값들의 집합이다.

18. 데이터베이스의 관계 모형에서 사용하는 테이블의 행을 구하는 애트리뷰트 값들의 집합을 무엇이라고 하는가?

- 가. DOMAIN    나. TUPLE    다. ENTITY    라. MEMBER

19. 관계 데이터베이스에서 릴레이션을 구성하고 있는 각각의 속성(Attribute)에서 취할 수 있는 원자 값들의 집합을 무엇이라 하는가?

- 가. 튜플(Tuple)
- 나. 도메인(Domain)
- 다. 개체 타입(Entity Type)
- 라. 개체 어커런스(Entity Occurrence)

20. 관계 데이터 모델에서 릴레이션(Relation)에 포함되어 있는 튜플(Tuple)의 수를 무엇이라고 하는가?

- 가. 차수(Degree)
- 나. 카디널리티(Cardinality)
- 다. 속성수(Attribute Value)
- 라. 카티션 프로덕트(Cartesian Product)

21. 데이터베이스 관리 시스템의 기능은 데이터를 정의하고 조작하며 제어하는 것이다. 정의 기능은 데이터베이스의 구조와 특성을 정의할 때는 데이터 모델에 따라 명세하고 정의한다. 데이터 모델 중에서 캡슐화(Capsulation), 상속(Inheritance), 다형성(Polymorphism)의 개념을 가지는 데이터 모델은?

- 가. 관계 데이터 모델(Relational Data Model)
- 나. 계층 데이터 모델(Hierarchical Data Model)
- 다. 네트워크 데이터 모델(Network Data Model)
- 라. 객체지향 데이터 모델(Object-Oriented Data Model)

[정답] 17.가 18.나 19.나 20.나 21.라

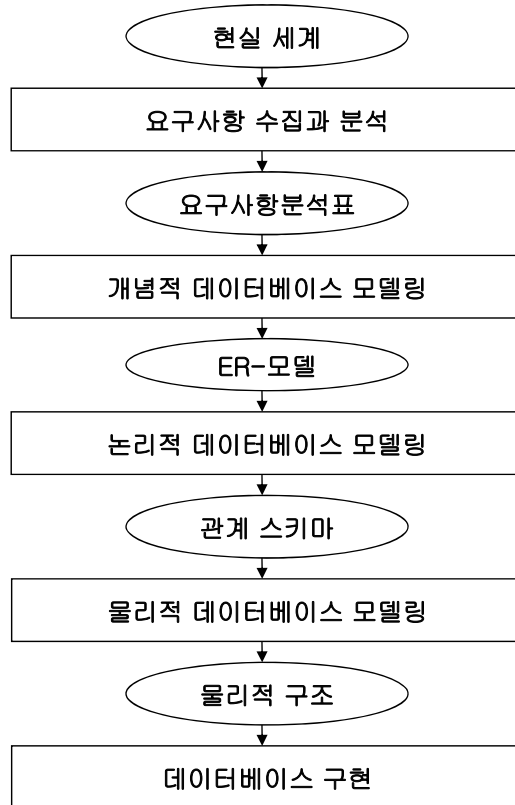
## 1. 관계형 데이터베이스 모델링

조직도/업무 분장표, 업무별 흐름도 →  
인터뷰 내용, 입출력 장표  
기존 시스템 분석, 새로운 요구사항

실세계의 정보 구조의 모형을 변환하여 →  
**일반화** 시키는 단계

목표 **DBMS 이론** 적용, **정규화** →

속성의 데이터타입과 사이즈 정의 →  
데이터 사용량 분석  
사용자들의 업무 프로세스 분석  
역정규화



## 1. 정규화 (normalization)

: 정규화를 하는 이유는 데이터의 **중복**을 방지하고 보다 효율적으로 데이터를 저장하기 위한.  
(릴레이션 분리 → 삽입, 삭제, 갱신 **이상**의 발생 가능성을 줄이는 것)

- 단점 : 연산 시간이 증가됨

## 2. 제1정규형 (1NF)

: 반복 되는 속성을 제거한 뒤 **모든 속성이 원자 도메인** 만으로 되어 있는 정규형

\* 제1정규형에 위배되는 테이블 (중복 발생 → 공간 낭비, 이상 발생 → 무결성 위배 가능)

[회원]

| 회원번호 | 이름  | 주소  | 이메일 | 주민번호 | 수강과목       | 과목내용   |
|------|-----|-----|-----|------|------------|--------|
| K001 | 이상희 | 서울시 | kkk | 111  | 정보처리<br>컴활 | A<br>B |
| K002 | 김시현 | 광주시 | ccc | 222  | 사무자동화      | C      |

[회원]

| 회원번호 | 이름  | 주소  | 이메일 | 주민번호 |
|------|-----|-----|-----|------|
| K001 | 이상희 | 서울시 | kkk | 111  |
| K002 | 김시현 | 광주시 | ccc | 222  |

[수강]

| 수강과목  | 회원번호 | 과목내용 |
|-------|------|------|
| 정보처리  | K001 | A    |
| 컴활    | K001 | B    |
| 사무자동화 | K002 | C    |

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 3. 이상(anomaly)

: 릴레이션에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하여 테이블 조작 시 불일치가 발생하는 것

- (1) 갱신 이상 : 반복된 데이터 중에 일부만 수정하면 데이터의 불일치가 발생  
→ 이상희라는 사람의 주소를 변경할 경우 모든 속성(칼럼)의 주소를 변경해야 한다.  
만약 하나만 변경할 경우 데이터의 불일치가 발생한다.
- (2) 삽입 이상 : 불필요한 정보를 함께 저장하지 않고는 어떤 정보를 저장하는 것이 불가능  
→ MOS라는 과목을 추가할 경우, 불필요한 회원정보까지 추가해야 한다.
- (3) 삭제 이상 : 유용한 정보를 함께 삭제하지 않고는 어떤 정보를 삭제하는 것이 불가능  
→ 이상희라는 사람의 데이터를 삭제하고자 할 경우, 정보처리라는 과목까지 삭제되어 버린다.

[회원]

| 회원번호 | 이름  | 주소  | 이메일 | 주민번호 | 수강과목  | 과목내용 |
|------|-----|-----|-----|------|-------|------|
| K001 | 이상희 | 서울시 | kkk | 111  | 정보처리  | A    |
| K001 | 이상희 | 서울시 | kkk | 111  | 컴활    | B    |
| K002 | 김시현 | 광주시 | ccc | 222  | 사무자동화 | C    |
| K002 | 김시현 | 광주시 | ccc | 222  | 컴활    | B    |
| K002 | 김시현 | 광주시 | ccc | 222  | 워드    | D    |

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 4. 제2정규형 (2NF)

: 제1정규형이고, **부분함수적 종속을 제거**하여 **완전(충분한) 함수적 종속**을 만족하는 정규형.

#### 1) 함수적 종속

- 어떤 릴레이션 R에서 X와 Y를 각각 R의 애트리뷰트 집합의 부분 집합이라고 할 경우, 애트리뷰트 X의 값 각각에 대하여 시간에 관계없이 항상 애트리뷰트 Y의 값이 오직 하나만 연관되어 있을때 Y는 X에 함수 종속이라 하고, X를 결정자, Y를 종속자 한다.

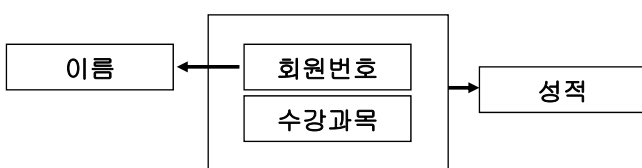
- 표기법 :  $X \rightarrow Y$

- 예) 회원번호  $\rightarrow$  이름

[회원]

| 회원번호 | 이름  | 주소  | 이메일 |
|------|-----|-----|-----|
| K001 | 이상희 | 서울시 | kkk |
| K002 | 김시현 | 광주시 | ccc |

#### 2) 부분 / 완전 함수적 종속



[성적]

| 회원번호 | 수강과목 | 성적  | 이름  |
|------|------|-----|-----|
| K001 | 정보필기 | 80  | 이상희 |
| K002 | 정보필기 | 90  | 김시현 |
| K002 | 정보실기 | 90  | 김시현 |
| K001 | 정보실기 | 100 | 이상희 |

**완전 함수적 종속** : 회원번호, 수강과목  $\rightarrow$  성적

**부분 함수적 종속** : 회원번호  $\rightarrow$  이름

이름 속성을 분리하면 성적 테이블은 완전함수적 종속 상태가 되므로 2정규형을 만족한다.



## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 5. 제3정규형 (3NF)

: 제2정규형이고, **이행적 함수적 종속 관계 제거**하여 **비이행적 함수적** 종속 관계를 만족하는 정규형

1) 이행적 함수적 종속

신청번호 → 회원번호, 회원번호 → 회원취미

: 신청번호 → 회원취미 는 이행적 함수적 종속

[신청]

| 신청번호 | 제품번호  | 회원번호 | 회원취미 |
|------|-------|------|------|
| 1    | A1111 | kkk  | 독서   |
| 2    | B2222 | qqq  | 영화   |
| 3    | C3333 | www  | 음악   |
| 4    | D4444 | eee  | 축구   |

회원번호, 회원취미 속성을 분리하면 신청 테이블은 비이행적 함수적 종속 상태가 되므로 3정규형을 만족한다.

### 6. BCNF (Boyce/Codd Normal Form)

: 제3정규형이고, **결정자가 후보키가 아닌 함수 종속 제거** 모든 **결정자**가 **후보키**이어야 한다는 것

[수강]

| 학번  | 과목    | 교수 |
|-----|-------|----|
| 001 | 프로그래밍 | A  |
| 001 | 자료구조  | B  |
| 002 | 프로그래밍 | A  |
| 002 | 자료구조  | C  |
| 003 | 자료구조  | C  |
| 003 | 프로그래밍 | D  |

후보키 : (학번, 과목), (학번, 교수)

함수적 종속 : (학번, 과목) → 교수, 교수 → 과목

결정자 : (학번, 과목), 교수

교수 속성이 결정자이나 후보키가 아니므로 교수 속성을 분리하면 수강 테이블은 모든 결정자가 후보키이므로 BCNF를 만족한다.

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 7. 제4정규형 (4NF)

: **다치 종속** 제거

- 키워드 : 다중값 종속(=다치 종속) 제거,  $A \twoheadrightarrow B$

- 다치 종속 :

먼저 함수적 종속의 개념을 아셔야 합니다.

릴레이션에서 아이디 속성이 있고, 주민번호 속성이 있다고 할때, 아이디 속성을 알면 주민번호 속성을 알 수 있습니다.(아이디 속성은 주민번호 속성을 결정하는 결정자입니다.)

이 때 아이디→주민번호 로 표현하고 함수적 종속이라고 합니다. 여기서 주의할 것은 1:1 로 대응되어야 합니다.

(하나의 아이디 값이 2개이상의 주민번호 속성을 결정하면 안됩니다.) 그럼 다치 종속은 설명이 쉽습니다. 1:

다 대응을 생각하시면 됩니다. 릴레이션에서 아이디 속성이 있고, 수강과목 속성이 있다면 하나의 아이디는 여러개 과목을 수강할 수 있으므로 아이디와 수강과목 속성은 함수적 종속에서 다치 종속이라고 합니다. 표현은 아이디→>수강과목 입니다. 이런 다치 종속은 이상이 발생할 수 있으므로 무손실 분해(정규화)되어야 합니다. 이것을 4정규형이라고 합니다.

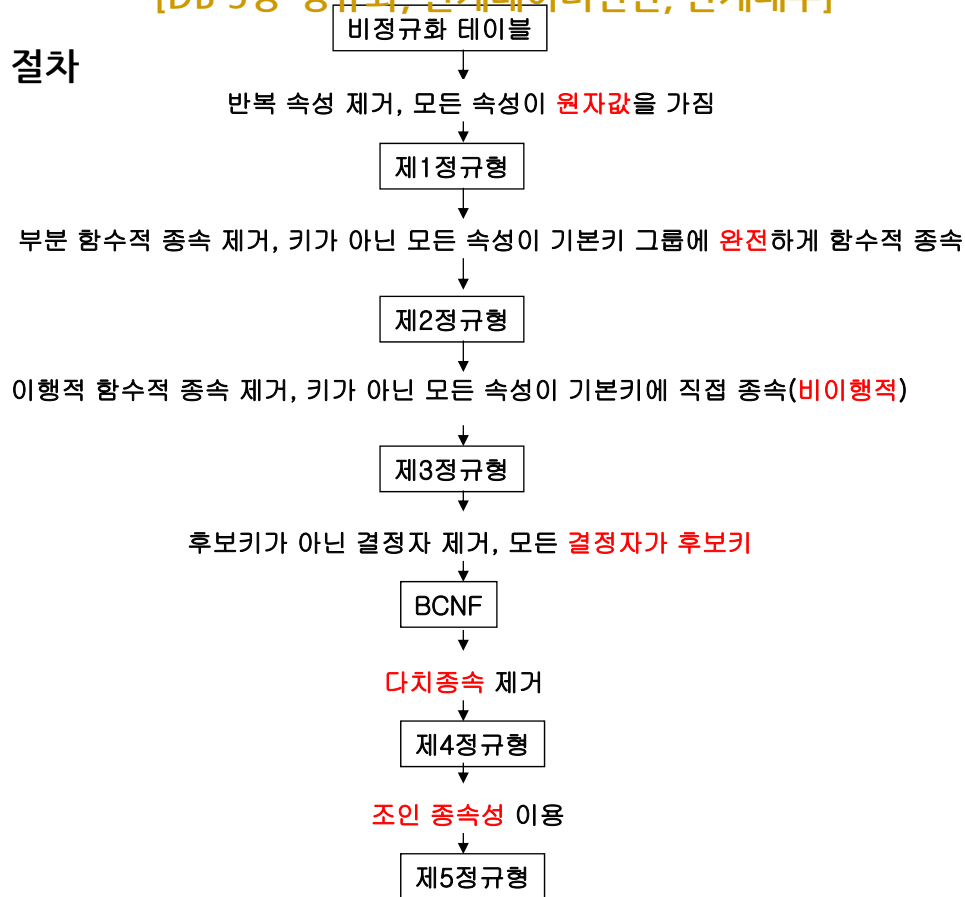
### 8. 제5정규형 (5NF)

: **조인 종속성** 이용



## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 9. 정규화 절차



## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 10. 정규화 특징

- 예를 들어 현재 테이블이 3정규형 상태라면 1,2 정규형은 자동으로 만족한다.
- 정규형들은 차수가 높아질수록(제1정규형 → 제5정규형)만족시켜야 할 제약조건이 증가된다.
- 정규화는 논리적 처리 및 품질에 큰 영향을 미친다.
- 정규화의 목적은 논리적 데이터베이스 구조상에 있어 삽입, 수정, 그리고 삭제 결과 생기는 이상현상(anomaly)을 제거하는데 있다.
- 레코드들의 관련 속성들 간의 종속성을 최소화하기 위한 구성 기법이다
- 정규화가 잘못되면 데이터의 불필요한 중복을 야기하여 릴레이션 조작시 문제를 일으킨다.
- 정규화되지 못한 릴레이션의 조작시 발생하는 이상(anomaly) 현상의 근본적인 원인은 여러 가지 종류의 사실들이 하나의 릴레이션에 표현되기 때문이다.

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 1. 관계 데이터 연산 ★★☆☆☆

| 관계 대수   | 관계 해석   |
|---|---|
| 1. <b>절차적</b> 언어(절차 중심)<br>- 원하는 정보를 ' <u>어떻게</u> ' 유도하는가를 연산자와 연산규칙 이용하여 기술<br><br>2. 분류 : 순수 관계 연산자, 일반 집합 연산자<br><br>3. SQL의 이론적인 기초 | 1. <b>비절차적</b> 언어(결과 중심)<br>- 원하는 정보가 ' <u>무엇</u> '이라는 것만 정의<br><br>2. 분류 : 튜플 관계 해석과 도메인 관계 해석 |

- 기본적으로 관계 해석과 관계 대수는 관계 데이터베이스를 처리하는 기능과 능력면에서 동등하다.
- 관계 해석으로 표현한 식은 관계대수로 표현할 수 있다.

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 1. 관계 대수 종류 ★★☆☆☆

- 순수 관계 연산자 : SELECT, PROJECT, JOIN, DIVISION
- 일반 집합 연산자 : 합집합, 교집합, 차집합, 카티션프로덕트(곱하기 연산)

### 2. SELECT ( $\sigma$ )

- 릴레이션에서 주어진 조건을 만족하는 **튜플**들을 검색하는 것으로 기호는 그리스 문자의  **$\sigma$**  (시그마)를 이용한다. (**행, 수평적 연산**)

- 형식 :  $\sigma$  조건 (R)

수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 | 정보 | 서구 | 120 |
| 남기욱 | 정보 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |



| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |

$\sigma$  이름 = '김길현' (**수강생**)

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 3. PROJECT ( $\pi$ )

- 릴레이션에서 주어진 조건을 만족하는 속성들을 검색하는 것으로, 기호는 그리스 문자의 **파이**( $\pi$ )를 이용한다. (**열**, 수직적 연산)

- 형식 :  $\pi$  속성 (R)

수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 | 정보 | 서구 | 120 |
| 남기욱 | 정보 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |



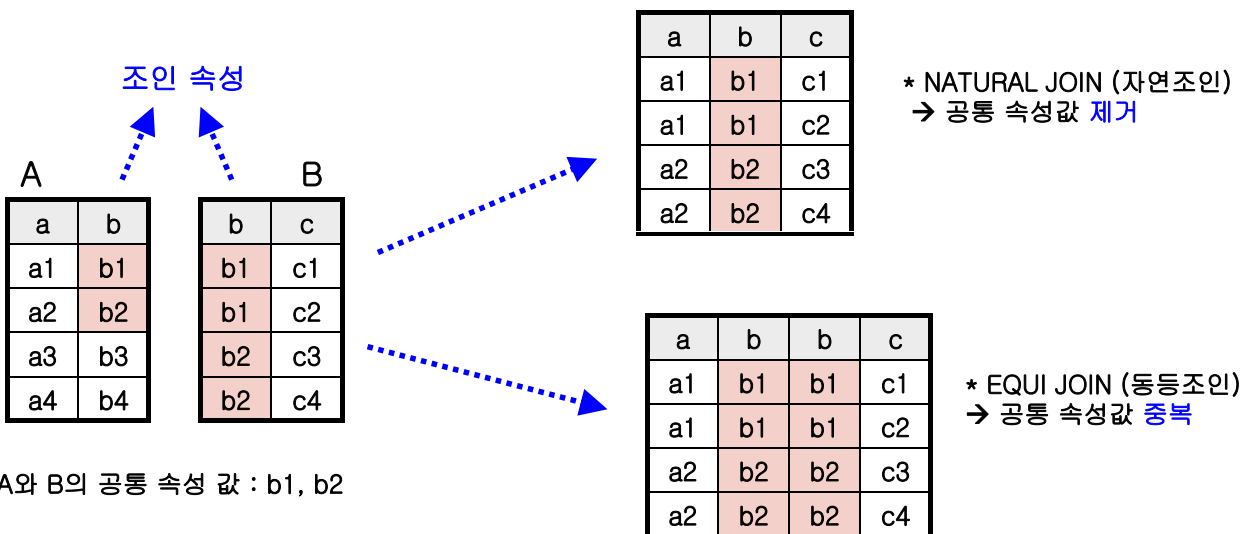
| 이름  |
|-----|
| 김길현 |
| 이상인 |
| 남기욱 |
| 최영희 |
| 김상현 |

$\pi$  이름 (수강생)

## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 4. JOIN ( $\bowtie$ )

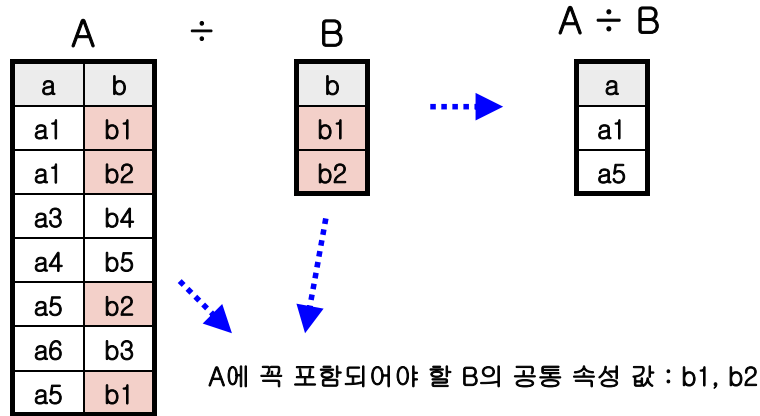
- 두 개의 릴레이션 A와 B에서 공통된 속성을 연결하는 것이다.



## [DB 5강-정규화, 관계데이터연산, 관계대수]

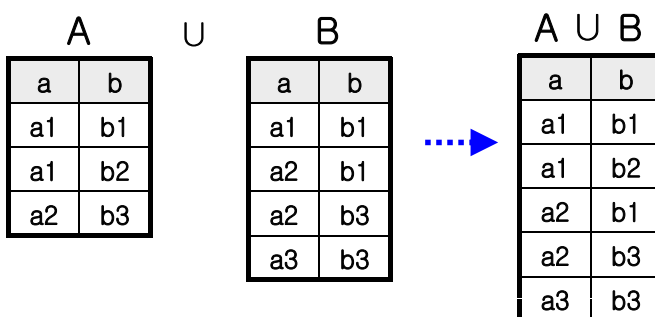
### 5. DIVISION ( $\div$ )

- 나누어지는 릴레이션인 A는 릴레이션 B의 모든 내용을 포함한 것이 결과 릴레이션이 된다

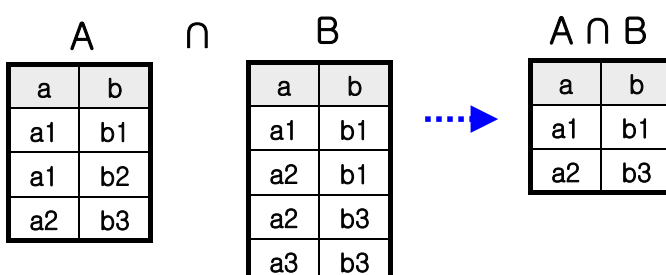


## [DB 5강-정규화, 관계데이터연산, 관계대수]

### 6. 합집합 ( $\cup$ ) : 릴레이션 A 또는 B에 속하는 튜플들로 구성된 릴레이션이다. (UNION)



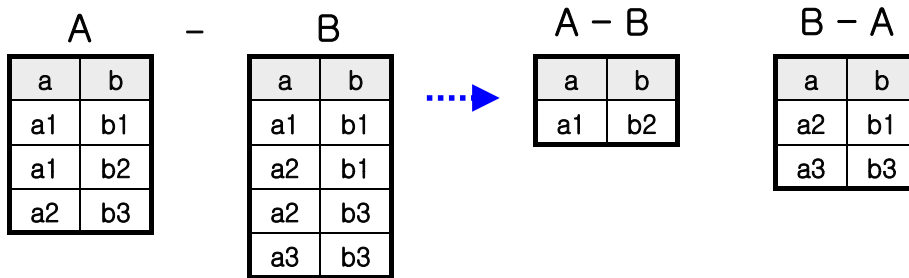
### 7. 교집합 ( $\cap$ ) : 릴레이션 A 와 B에 공통적으로 속하는 튜플들로 구성된 릴레이션이다. (INTERSECTION)



관계 대수 연산자 중 **합집합**, **교집합**, **차집합** 연산은 이항 연산으로서 연산에 참가하는 두 개의 릴레이션은 차수와 도메인이 같아야 연산을 수행할 수 있다.

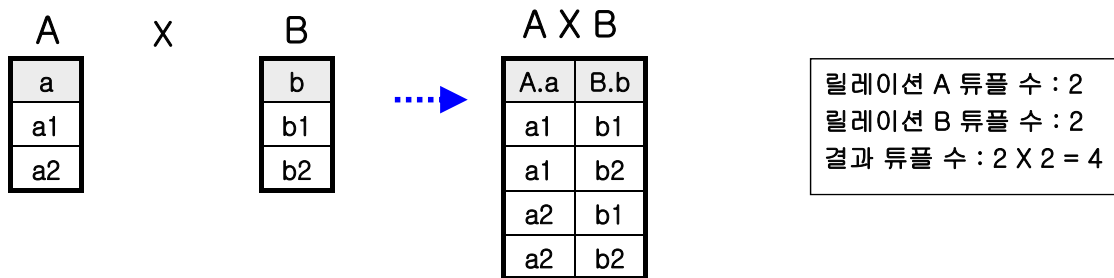
## [DB 5강-정규화, 관계데이터연산, 관계대수]

**8. 차집합 (-)** : 릴레이션 A에만 있고 B에는 없는 튜플들로 구성된 릴레이션이다. (DIFFERENCE)



## 9. 카티션 프로덕트(cartesian product) (X)

- A에 속한 각 튜플 a에 대하여 B에 속한 튜플 b를 모두 접속시킨 튜플들(a b)로 구성된 릴레이션이다.



## [DB 5강-정규화, 관계데이터연산, 관계대수]

1. 조건을 만족하는 릴레이션의 수평적 부분집합으로 구성하며, 연산자의 기호는 그리스 문자 시그마( $\sigma$ )를 사용하는 관계대수 연산자는?

- 가. Select 연산자      나. Project 연산자  
 다. Join 연산자      라. Division 연산자

2. 관계 데이터베이스에 적용할 순수 관계 연산자로 거리가 먼 것은?

- 가. 링크(Link)      나. 선택트(Select)  
 다. 디비전(Division)      라. 프로젝트(Project)

3. 정규화의 의미로 틀린 것은?

- 가. 함수적 종속성 등의 종속성 이론을 이용하여 잘못 설계된 관계형 스키마를 더 작은 속성의 세트로 쪼개어 바람직한 스키마로 만들어 가는 과정이다.  
 나. 좋은 데이터베이스 스키마를 생성해 내고 불필요한 데이터의 중복을 방지하여 정보 검색을 용이하게 해준다.  
 다. 정규형에는 제1정규형, 제2정규형, 제3정규형, BCNF형, 제4정규형, 제5정규형 등이 있다.  
 라. 어떠한 Relation 구조가 바람직한 것인지, 바람직하지 못한 Relation을 어떻게 합쳐야 하는지에 관한 구체적인 판단 기준을 제공한다.

4. 다음 관계 대수 문장의 의미는?

$\Pi$  이름( $\sigma$ 학과='전산'(교수))

- 가. 전산학과 교수들의 이름을 검색하시오  
 나. 전산학과 교수들의 이름 테이블을 삭제하시오  
 다. 전산학과 교수들의 이름을 삭제하시오  
 라. 전산학과 교수들의 이름을 삽입하시오

5. 릴레이션 R에는 10개의 튜플이 있고, 다른 릴레이션 S에는 5개의 튜플이 있을 때, 두 개의 릴레이션 R과 S의 교차곱(Cartesian Product) 연산을 수행한 후의 튜플의 수는?

- 가. 15개      나. 50개      다. 10개      라. 2개

6. 테이블에서 특정 속성에 해당하는 열을 선택하는데 사용되며 결과로는 릴레이션의 수직적 부분 집합에 해당하는 관계 대수 연산자는?

- 가. project 연산자  
 나. join 연산자  
 다. division 연산자  
 라. select 연산자

[정답] 1.가 2.가 3.라 4.가 5.나 6.가

## [DB 5강-정규화, 관계데이터연산, 관계대수]

7. 관계형 데이터베이스의 릴레이션을 조작할 때 발생하는 이상(Anomaly) 현상에 관한 설명으로 적절하지 않은 것은?

- 가. 데이터의 종속으로 인해 발생하는 이상 현상에는 삭제 이상, 삽입 이상, 갱신 이상이 있다.
- 나. 릴레이션의 한 튜플을 삭제함으로써 연쇄 삭제로 인해 정보의 손실을 발생시키는 현상이 삭제 이상이다.
- 다. 데이터를 삽입할 때 불필요한 데이터가 함께 삽입되는 현상을 삽입 이상이라 한다.
- 라. 튜플 중에서 일부 속성을 갱신함으로써 정보의 모순성이 발생하는 현상이 갱신 이상이다.

8. 키가 아닌 모든 속성이 기본 키(Primary key)에 충분한 함수적 종속을 만족하는 정규형은?

- 가. 1NF    나. 2NF    다. 3NF    라. 4NF

9. 제3정규형에서 보이스코드 정규형(BCNF)으로 정규화하기 위한 작업은?

- 가. 원자값이 아닌 도메인을 분해
- 나. 부분 함수 종속 제거
- 다. 이행 함수 종속 제거
- 라. 결정자가 후보키가 아닌 함수 종속 제거

10. 어떤 릴레이션 R에 존재하는 모든 조인 종속성이 릴레이션 R의 후보키를 통해서만 성립된다. 이 릴레이션 R은 어떤 정규형의 릴레이션인가?

- 가. 제3정규형
- 나. 보이스-코드 정규형
- 다. 제4정규형
- 라. 제5정규형

11. 관계 데이터베이스에 있어서 관계 대수 연산이 아닌 것은?

- 가. 디비전(division)    나. 프로젝션(projection)
- 다. 조인(join)    라. 포크(fork)

12. 관계 해석(relational calculus)에 대한 설명으로 옳지 않은 것은?

- 가. 관계 해석으로 질의어를 표현한다.
- 나. 원하는 릴레이션을 정의하는 방법을 제공하며, 비절차적인 언어이다.
- 다. 튜플 관계 해석과 도메인 관계 해석이 있다.
- 라. 릴레이션 조작을 위한 연산의 집합이다.

[정답] 7.가 8.나 9.라 10.라 11.라 12.라

## [DB 5강-정규화, 관계데이터연산, 관계대수]

13. 어떤 릴레이션에 속한 모든 도메인이 원자값(atomic value)만으로 되어 있는 릴레이션을 무엇이라고 하는가?

- 가. 제 1 정규형(1NF)    나. 제 2 정규형(2NF)
- 다. BCNF    라. 제 4 정규형(4NF)

14. 정규화의 목적으로 거리가 먼 것은?

- 가. 삽입, 삭제, 갱신 이상의 발생을 방지한다.
- 나. 데이터의 중복성을 최소화 한다.
- 다. 효율적으로 데이터를 조작할 수 있다.
- 라. 릴레이션을 분해하여 연산시간을 감소시킨다.

15. 다음 중 SQL의 합집합 연산이 제대로 수행되는 경우는?

- 가. 두 테이블의 속성 개수가 같고, 대응되는 각 속성들의 도메인이 같으나 속성명들이 다른 경우
- 나. 두 테이블의 속성 개수가 같고, 대응되는 각 속성들의 도메인이 같으나 속성들의 도메인이 다른 경우
- 다. 두 테이블의 속성 개수가 다르나, 대응되는 각 속성들의 도메인이 같으며 속성명들이 같은 경우
- 라. 두 테이블의 속성 개수가 같으나 대응되는 각 속성들의 도메인이 다르고 속성명들이 다른 경우

16. 관계 해석(Relational Calculus)을 옳게 설명한 것은?

- 가. 연산들의 절차(sequence)를 사용하여 데이터를 가져온다.
- 나. 계산 수식을 사용하여 어떤 데이터를 가져올지 명시한다.
- 다. 기본적인 연산자로 UNION, INTERSECTION, DIFFERENCE를 사용한다.
- 라. 전체 관계를 조작하는데 사용되는 연산들의 집합이다.

17. 관계 대수(Relational Algebra)의 연산 중에서 두 릴레이션(Relation)의 교차 곱을 수행하기 때문에 두 릴레이션의 공통 튜플 수와 관계가 없는 것은?

- 가. UNION    나. INTERSECTION
- 다. DIFFERENCE    라. CARTESIAN PRODUCT

[정답] 13.가 14.라 15.가 16.나 17.라

## [DB 5강-정규화, 관계데이터연산, 관계대수]

18. 다음의 조건을 모두 만족하는 정규형은?

- 모든 도메인은 원자 값이고, 기본 키가 아닌 모든 속성들이 기본 키에 대해 완전 함수 종속적이며, 이행적 함수 종속 관계는 제거되었다.

가. 제 1 정규형  
다. 제 3 정규형

나. 제 2 정규형  
라. 제 1 정규형과 제 2 정규형

[정답] 18.다

## [DB 5강-정규화, 관계데이터연산, 관계대수]

[산-08년9월][산-08년5월]

19. 관계대수의 프로젝트 연산의 연산자 기호는?

가.  $\pi$   
다.  $\div$

나.  $\cap$   
라.  $\cup$

[기-08년5월]

20. 관계데이터베이스의 정규화에 대한 설명으로 옳지 않은 것은?

- 가. 정규화를 거치지 않으면 여러 가지 상이한 종류의 정보를 하나의 릴레이션으로 표현하여 그 릴레이션을 조작할 때 이상(Anomaly) 현상이 발생할 수 있다.
- 나. 정규화의 목적은 각 릴레이션에 분산된 종속성을 하나의 릴레이션에 통합하는 것이다.
- 다. 이상(Anomaly) 현상은 데이터들 간에 존재하는 함수종속이 하나의 원인이 될 수 있다.
- 라. 정규화가 잘못되면 데이터의 불필요한 중복이 야기되어 릴레이션을 조작할 때 문제가 발생할 수 있다.

[기-08년9월]

21. 정규화 과정 중 1NF에서 2NF가 되기 위한 조건은?

- 가. 1NF를 만족하고 모든 도메인이 원자 값이어야 한다.
- 나. 1NF를 만족하고 키가 아닌 모든 애트리뷰트들이 기본키에 이행적으로 함수 종속되지 않아야 한다.
- 다. 1NF를 만족하고 다치 종속이 제거되어야 한다.
- 라. 1NF를 만족하고 키가 아닌 모든 속성이 기본키에 완전 함수적 종속되어야 한다.

[정답] 19.가 20.나 21.라

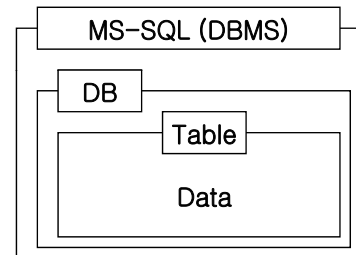
## 1. SQL(Structured Query Language) 특징



- 1) 관계대수와 관계해석을 기초로 한 고급 데이터 언어
- 2) 이해하기 쉬운 형태
- 3) 대화식 질의어로 사용 가능
- 4) 데이터 정의, 조작, 제어 기능 제공
- 5) COBOL, C, PASCAL 등의 언어에 삽입 -> 내장 SQL
- 6) 레코드 집합 단위로 처리
- 7) DBMS에서 사용되는 비 절차적 대화형 Language

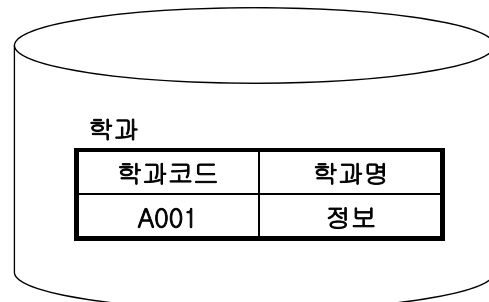
사용자

SQL



학과

|                |
|----------------|
| 학과코드: CHAR(18) |
| 학과명: CHAR(18)  |



[물리적 설계]

[데이터베이스 구현]

## 2. SQL 개념 이해하기 (실습)

- 1) 테이블을 생성 합니다.

**CREATE TABLE** 학과 (학과코드 CHAR(18), 학과명 CHAR(18));

1

| 학과코드 | 학과명 |
|------|-----|
|------|-----|

- 2) 자료(튜플)를 입력 합니다.

**INSERT INTO** 학과 **VALUES** ('A001','정보');

2,3

| 학과코드 | 학과명 |
|------|-----|
| A001 | 정보  |

- 3) 자료(튜플)를 검색 합니다.

**SELECT \* FROM** 학과;

4

| 학과코드 | 학과명 |
|------|-----|
| A001 | 사무  |

- 4) 자료(튜플)를 수정 합니다.

**UPDATE** 학과 **SET** 학과명='사무' **WHERE** 학과코드='A001';

- 5) 자료(튜플)를 삭제 합니다.

**DELETE FROM** 학과;

5

| 학과코드 | 학과명 |
|------|-----|
|------|-----|

- 6) 테이블을 삭제 합니다.

**DROP TABLE** 학과;





## [DB 6강-SQL, 시스템카탈로그, 뷰]

### CREATE VIEW

```
CREATE VIEW 여학생_view(이름, 성별) AS
SELECT 이름,성별
FROM 학생
WHERE 성별='여';
```

- \* 학생 테이블에서 조회가 빈번하게 발생하는 이름, 성별 (여학생) 을 뷰로 정의해서 시스템 성능을 향상시킴.
- \* AS : '별칭' 의 의미

| 학생   |     |    |           |      | 여학생_view |    |
|------|-----|----|-----------|------|----------|----|
| 학번   | 이름  | 성별 | 생년월일      | 학과코드 | 이름       | 성별 |
| A001 | 김기영 | 남  | 78년10월04일 | 사무   | 최현주      | 여  |
| A002 | 최현주 | 여  | 79년04월09일 | 정보   | 이미영      | 여  |
| A003 | 이미영 | 여  | 82년12월12일 | 사무   |          |    |

- \* **VIEW 정의** : 사용자에게 접근이 허용된 자료만을 제한적으로 보여주기 위해서 하나 이상의 기본 테이블로부터 유도된 **가상 테이블 (물리적 X, 논리적 O)**

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### \* VIEW 특징

- 구조가 기본테이블과 거의 유사
- 물리적으로 구현되지 않았다.
- 논리적 독립성 제공
- 필요한 데이터로만 구성 → 관리 수월, 명령 간단
- 데이터 보호 효율적 → 자동 보안
- 삽입, 삭제, 갱신 연산이 가능하지만 제한적이다.
- 다른 VIEW 정의에 기초
- 하나의 VIEW를 삭제 → 그 VIEW를 기초로 만들어진 VIEW도 자동 삭제
- 독립적인 인덱스를 가질 수 없다.
- 뷰에 대한 검색은 일반 테이블과는 같다.
- VIEW의 정의 변경(Alter VIEW) 불가

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### ALTER TABLE

ALTER TABLE **학과** ADD 연락처 CHAR(18);



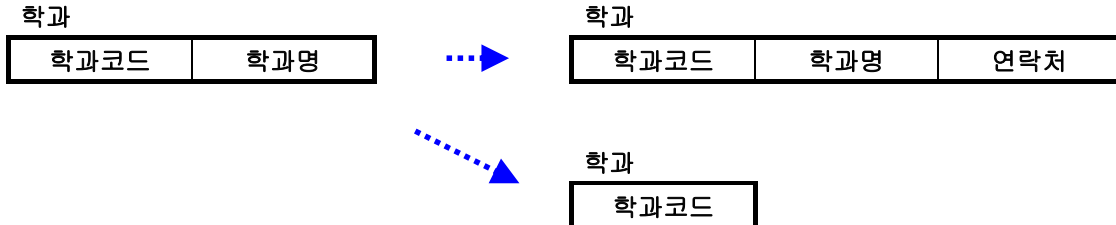
새로운 속성 추가

ALTER TABLE **학과** DROP 학과명;



속성 제거

\* 테이블의 구조를 변경 → 속성 변경



### DROP TABLE

DROP TABLE **학과** CASCADE;

\* 참조 무결성 위배를 피하기 위한 2가지 옵션

- CASCADE : 참조하는 테이블을 연쇄적으로 제거
- RESTRICT : 참조하는 테이블이 있을 경우 제거 안 됨 (생략 가능)

→ DELETE(삭제), UPDATE(갱신) 명령어도 옵션 사용

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### 6. DML (데이터 조작어)

: 데이터베이스에 저장된 자료(튜플)를 검색(SELECT), 삽입(INSERT), 삭제(DELETE), 갱신(UPDATE)하기 위한 언어

### INSERT

① INSERT INTO **수강생** VALUES ('김길현','정보','남구',100);

→ 수강생 테이블에 김길현, 정보, 남구, 100 을 넣어라.

② INSERT INTO **수강생**(이름,수강료) VALUES ('이상인',120);

→ 수강생 테이블에 이름, 수강료 에 이상인, 120을 넣어라.

③ INSERT INTO **정보수강생**(이름,과목,수강료)

SELECT 이름,과목,수강료 FROM **수강생** WHERE 주소='남구';

→ 수강생 테이블에서 주소가 남구인 이름, 과목, 수강료를 SELECT 해서 정보수강생 테이블 속성인 이름, 과목, 수강료에 넣어라.

③ 정보수강생

| 이름  | 과목 | 수강료 |
|-----|----|-----|
| 김길현 | 정보 | 100 |

수강생

| 이름 | 과목 | 주소 | 수강료 |
|----|----|----|-----|
|----|----|----|-----|



수강생

|   | 이름  | 과목 | 주소 | 수강료 |
|---|-----|----|----|-----|
| ① | 김길현 | 정보 | 남구 | 100 |
| ② | 이상인 |    |    | 120 |

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### DELETE

- ① DELETE FROM 수강생 WHERE 과목='사무';  
→ 수강생 테이블에서 과목이 사무인 튜플을 삭제하라.
  - ② DELETE FROM 수강생;  
→ 수강생 테이블에 있는 모든 튜플을 삭제하라.
- \* DELETE 는 튜플을 삭제하는 명령어 (테이블은 DROP)

수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 남기욱 | 사무 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |
| 이훈석 | 정보 | 동구 | 40  |
| 장기영 | 사무 |    | 100 |

① 수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |
| 이훈석 | 정보 | 동구 | 40  |

② 수강생

| 이름 | 과목 | 주소 | 수강료 |
|----|----|----|-----|
|----|----|----|-----|

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### UPDATE

- ① UPDATE 수강생 SET 과목='사무' WHERE 이름='최영희';  
→ 수강생 테이블에서 이름의 최영희인 튜플의 과목을 사무로 바꾸어라.
- ② UPDATE 수강생 SET 수강료=수강료+10 WHERE 과목='워드';  
→ 수강생 테이블에서 과목이 워드인 튜플의 수강료를 +10 해라.

수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 남기욱 | 사무 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |

① 수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 남기욱 | 사무 | 서구 | 100 |
| 최영희 | 사무 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |

② 수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 남기욱 | 사무 | 서구 | 100 |
| 최영희 | 사무 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 60  |

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### SELECT (기본구조)

SELECT      속성 → 속성 외에 식이나 그룹함수가 올 수 있다.  
FROM        테이블  
[WHERE      조건]; → [ ] 생략 가능

### SELECT

SELECT \* FROM 수강생;  
SELECT 수강생.\* FROM 수강생;  
SELECT 이름,과목,주소,수강료 FROM 수강생;  
SELECT 수강생.이름,수강생.과목,수강생.주소,수강생.수강료 FROM 수강생;

→ 같은 결과입니다. (모든 튜플 검색하라.)

수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 남기욱 | 사무 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |



| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 |    |    | 120 |
| 남기욱 | 사무 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |

## [DB 6강-SQL, 시스템카탈로그, 뷰]

- ① SELECT DISTINCT 과목 FROM 수강생;  
→ 수강생 테이블에서 중복 제거 된 과목만 검색하시오.
- ② SELECT \* FROM 수강생 WHERE 과목='정보';  
→ 수강생 테이블에서 과목이 정보인 모든 튜플을 검색하시오.
- ③ SELECT \* FROM 수강생 WHERE 과목='컴활' OR 과목='워드';  
→ 수강생 테이블에서 과목이 컴활 이거나 워드인 모든 튜플을 검색하시오.

수강생

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 | 정보 | 서구 | 120 |
| 남기욱 | 정보 | 서구 | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |



①

| 과목 |
|----|
| 정보 |
| 컴활 |
| 워드 |



②

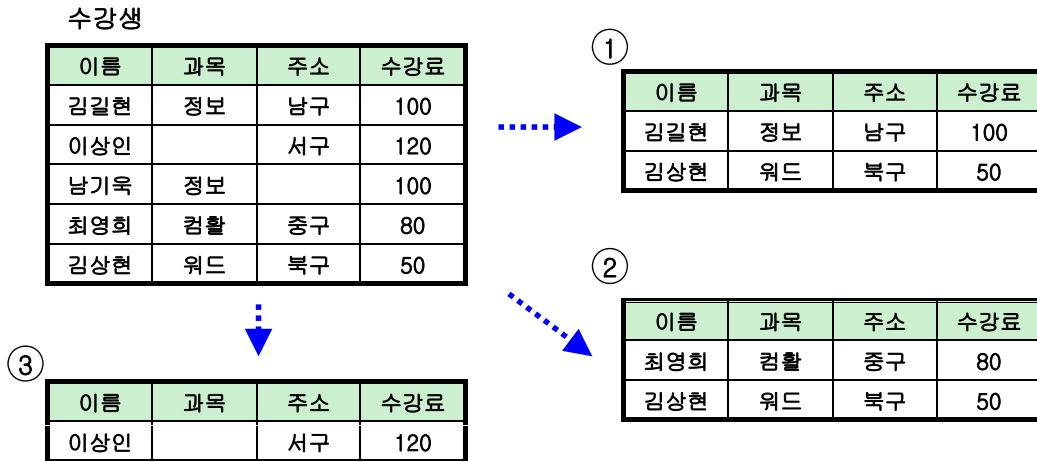
| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 | 정보 | 서구 | 120 |
| 남기욱 | 정보 | 서구 | 100 |

③

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |

## [DB 6강-SQL, 시스템카탈로그, 뷰]

- ① SELECT \* FROM 수강생 WHERE 이름 LIKE '김%';  
→ 수강생 테이블에서 이름이 '김'으로 시작하는 모든 튜플을 검색하시오.
- ② SELECT \* FROM 수강생 WHERE 수강료 BETWEEN 30 AND 90;  
→ 수강생 테이블에서 수강료가 30이상 90이하인 모든 튜플을 검색하시오.
- ③ SELECT \* FROM 수강생 WHERE 과목 IS NULL;  
→ 수강생 테이블에서 과목이 NULL인 모든 튜플을 검색하시오. (주의 : CREATE 문은 IS 를 생략)



## [DB 6강-SQL, 시스템카탈로그, 뷰]

### SELECT (확장구조)

```
SELECT [DISTINCT] 속성
FROM   테이블
[WHERE 조건]
[GROUP BY 속성 [HAVING 조건]]
[ORDER BY 속성 [ASC | DESC]];
```

- \* **DISTINCT** : 검색 결과에서 중복 배제 (주의 : CREATE 문의 UNIQUE 과 구분)
- \* **GROUP BY** : 그룹별 검색 (예, 과목별 수강생 수를 구하라.)
- \* **ORDER BY** : 정렬 검색 (오름차순 : A-Z, ↗-↘)

### 그룹 함수

```
COUNT(속성) : 그룹별 튜플 수
AVG(속성) : 그룹별 평균
SUM(속성) : 그룹별 합계
MAX(속성) : 그룹별 최대값
MIN(속성) : 그룹별 최소값
```

## [DB 6강-SQL, 시스템카탈로그, 뷰]

- ① SELECT \* FROM 수강생 WHERE 과목='정보' ORDER BY 이름 DESC;  
→ 수강생 테이블에서 과목이 정보인 튜플을 검색하되, 이름을 기준으로 내림차순 정렬하여 모든 튜플을 검색하시오.  
(정렬 방법을 명시하지 않을 경우 기본적으로 ASC (오름차순) 정렬됨.)
- ② SELECT 과목, COUNT(\*) AS 수강생수 FROM 수강생 WHERE 수강료>=100  
GROUP BY 과목 HAVING COUNT(\*)>=2;  
→ 수강생 테이블에서 수강료가 100이상인 과목별 수강생이 2명 이상인 수강생수를 검색하시오.

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 김길현 | 정보 | 남구 | 100 |
| 이상인 | 컴활 | 서구 | 120 |
| 남기욱 | 정보 |    | 100 |
| 최영희 | 컴활 | 중구 | 80  |
| 김상현 | 워드 | 북구 | 50  |

①

| 이름  | 과목 | 주소 | 수강료 |
|-----|----|----|-----|
| 남기욱 | 정보 |    | 100 |
| 김길현 | 정보 | 남구 | 100 |

②

| 과목 | 수강생수 |
|----|------|
| 정보 | 2    |

\* SQL 수행 순서 : FROM → WHERE → GROUP BY → HAVING → SELECT → ORDER BY

## [DB 6강-SQL, 시스템카탈로그, 뷰]

SELECT \* FROM 수강생 UNION  
SELECT \* FROM 정회원;  
→ 수강생 테이블과 정회원 테이블의 모든 튜플을 중복 없이 하나의 결과로 병합 검색하라.

| 이름  | 과목 |
|-----|----|
| 김길현 | 정보 |
| 이상인 | 컴활 |
| 남기욱 | 정보 |
| 최영희 | 컴활 |
| 김상현 | 워드 |

.....→

| 이름  | 과목 |
|-----|----|
| 김길현 | 정보 |
| 최기영 | 컴활 |
| 남기욱 | 정보 |
| 최영희 | 컴활 |
| 송기성 | 워드 |

.....→

| 이름  | 과목 |
|-----|----|
| 김길현 | 정보 |
| 이상인 | 컴활 |
| 남기욱 | 정보 |
| 최영희 | 컴활 |
| 김상현 | 워드 |
| 최기영 | 컴활 |
| 송기성 | 워드 |

\* 관계대수를 SQL 로 변환하기

$\pi$  이름, 학년 ( $\sigma$  학과 = '컴퓨터' (학생))

→ SELECT 이름, 학년 FROM 학생 WHERE 학과 = '컴퓨터';

## [DB 6강-SQL, 시스템카탈로그, 뷰]

### 7. DCL (데이터 제어어)

- : DBA가 데이터 관리를 목적으로 사용하는 언어로서 보안,회복,사용자 권한 등을 정의 한다.
- GRANT (권한부여), REVOKE(권한취소)
  - COMMIT(트랜잭션 완료->DB 적용), ROLLBACK(트랜잭션 취소->DB 적용 안됨)

#### GRANT/REVOKE

##### \* 권한 부여

GRANT SELECT ON 수강생 TO KWS;

→ KWS 에게 수강생 테이블에 대한 SELECT 권한 부여 (KWS는 다른 사람에게 권한을 부여할 수 없음)

→ 권한 : ALL, INSERT, DELETE, UPDATE, SELECT 등

GRANT SELECT ON 수강생 TO KWS WITH GRANT OPTION;

→ WITH GRANT OPTION (KWS는 다른 사람에게 권한을 부여할 수 있음)

##### \* 권한 취소

REVOKE SELECT ON 수강생 FROM KWS CASCADE;

→ KWS 로 부터 수강생 테이블에 대한 SELECT 권한을 취소 (KWS로 부터 권한을 부여 받은 사용자도 연쇄적으로 취소)

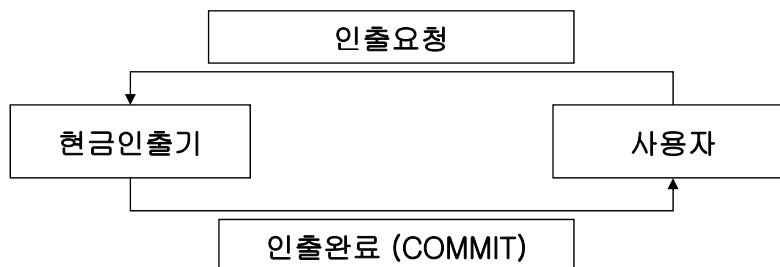
## [DB 6강-SQL, 시스템카탈로그, 뷰]

#### COMMIT/ROLLBACK

- \* COMMIT : 트랜잭션의 성공했을 경우 그 결과를 DB에 적용하여 완료 시킴.
- \* ROLLBACK : 트랜잭션의 실패로 작업을 취소하고, 이전 상태로 되돌림.

확인

취소



- \* 인출과정 전체를 Transaction 이라고 한다. → 작업의 논리적인 단위
- \* 모든 작업이 성공한 경우 Commit 을 해주고, 중간에 조금이라도 실수가 있었다면 Roll Back 을 하게 됩니다.
- \* 롤백을 하면 Transaction 을 하기 전까지의 상태로 돌릴 수가 있습니다.



## [DB 6강-SQL, 시스템카탈로그, 뷰]

SELECT 과목, COUNT(\*) AS 수강생수 FROM 수강생 WHERE 수강료>=100

GROUP BY 과목 HAVING COUNT(\*)>=2;

→ 수강생 테이블에서 수강료가 100이상인 과목별 수강생이 2명 이상인 수강생수를 검색하시오.

[해석 순서]

1. where 조건 처리 : 조건에 맞는 튜플을 작업 대상으로 정하고,
2. 그룹된 속성의 조건 처리 : 과목별 건수가 2 이상인 조건 처리
3. select 과목,수강생수 → 수강생수는 건수가 들어 갑니다.

count(\*) : 튜플의 건수 → Null 포함

count(속성명) : 튜플의 건수 → Null 제외

count(DISTINCT 속성명) : NULL, 중복 제외된 건수 반환

[AS]

CREATE VIEW 뷰명(속성명) AS SELECT문;

[ON, TO]

CREATE [UNIQUE] INDEX 인덱스명 ON 테이블명(속성명);

GRANT 사용자등급 ON 테이블 TO 사용자\_ID;

[IN]

SELECT 속성명 FROM 테이블명 WHERE 속성명 IN (SELECT문); → IN 또는 NOT IN

## [DB 6강-SQL, 시스템카탈로그, 뷰]

1. SQL 언어에 포함되는 기능이 아닌 것은?

가. DCL 나. DDL 다. DQL 라. DML

2. SQL 언어의 데이터 제어어(DCL)에 해당하는 것은?

가. SELECT문 나. INSERT문  
다. UPDATE문 라. GRANT문

3. SQL 구문과 의미가 잘못 연결된 것은?

가. CREATE - 테이블 생성  
나. DROP - 레코드 삭제  
다. UPDATE - 자료 갱신  
라. DESC - 내림차순 정렬

4. 다음 SQL문의 실행 결과를 가장 올바르게 설명한 것은?

Drop Table 인사 Cascade;

가. 인사 테이블을 제거한다.  
나. 인사 테이블을 참조하는 테이블과 인사 테이블을 제거한다.  
다. 인사 테이블이 참조 중이면 제거하지 않는다.  
라. 인사 테이블을 제거할지의 여부를 사용자에게 다시 질의한다.

5. 다음 [ ]에 적당한 SQL 문장은?

SELECT 번호, 이름  
FROM 학생테이블  
[ ]학과 = "컴퓨터";

가. SET 나. GROUP 다. WITH 라. WHERE

6. 다음 SQL문에서 WHERE절의 조건이 의미하는 것은?

SELECT CNO, CNAME  
FROM COURSE  
WHERE CNO LIKE 'S\_ \_';

가. S로 시작되는 3문자의 CNO를 검색한다.  
나. S로 시작되는 모든 문자 CNO를 검색한다.  
다. 문자열로만 이루어진 모든 CNO를 검색한다.  
라. S를 포함한 모든 CNO를 검색한다.

[정답] 1.다 2.라 3.나 4.나 5.라 6.가

## [DB 6강-SQL, 시스템카탈로그, 뷰]

7. 다음 릴레이션 R1과 R2에 대해 아래의 SQL문을 실행한 결과는?

```
SELECT B
FROM R1
WHERE C = (SELECT C FROM R2 WHERE D = 'K');
```

| R1 |   |   | R2 |   |   |
|----|---|---|----|---|---|
| A  | B | C | C  | D | E |
| 1  | a | x | x  | k | 3 |
| 2  | b | x | y  | x | 3 |
| 1  | c | y | z  | 1 | 2 |

가. a    나. b    다. a b c    라. a b

8. SQL의 기술이 옳지 않은 것은?

- 가. SELECT...FROM...WHERE...
- 나. INSERT...INTO...VALUES...
- 다. UPDATE...TO...WHERE...
- 라. DELETE...FROM...WHERE...

9. SQL에서 각 기능에 대한 내장함수의 연결이 옳지 않은 것은?

- 가. 열에 있는 값들의 개수 - COUNT
- 나. 열에 있는 값들의 평균 - AVG
- 다. 열에 있는 값들의 합 - TOT
- 라. 열에서 가장 큰 값 - MAX

10. 학생(STUDENT) 테이블에 전산과 학생이 50명, 경영과 학생이 100명, 기계과 학생이 50명 있다고 할 때, 다음 SQL문 ㉠, ㉡, ㉢의 실행 결과 튜플수는 각각 얼마인가? (단, DEPT 필드는 학과명을 의미한다.)

```
㉠ SELECT DEPT FROM STUDENT
㉡ SELECT DISTINCT DEPT FROM STUDENT
㉢ SELECT COUNT(DISTINCT DEPT)
    FROM STUDENT WHERE DEPT = '전산과'
```

- 가. ㉠ 3    ㉡ 3    ㉢ 1
- 나. ㉠ 200    ㉡ 3    ㉢ 1
- 다. ㉠ 200    ㉡ 3    ㉢ 50
- 라. ㉠ 200    ㉡ 200    ㉢ 50

11. SQL문에서 HAVING을 사용할 수 있는 절은?

- 가. LIKE 절
- 나. WHERE 절
- 다. GROUP BY 절
- 라. ORDER BY 절

[정답] 7.라 8.다 9.다 10.나 11.다

## [DB 6강-SQL, 시스템카탈로그, 뷰]

12. 다음 표와 같은 판매 실적 테이블을 읽어 서울 지역에 한하여 판매액 내림차순으로 지점명과 판매액을 출력하고자 한다. 가장 적절한 SQL 구문은?

<판매 실적 테이블>

| 도시 | 지점명  | 판매액 |
|----|------|-----|
| 서울 | 강남지점 | 330 |
| 서울 | 강북지점 | 168 |
| 광주 | 광주지점 | 197 |
| 서울 | 강서지점 | 158 |
| 서울 | 강동지점 | 197 |
| 대전 | 대전지점 | 165 |

- 가. SELECT 지점명, 판매액 FROM 판매실적 WHERE 도시='서울' ORDER BY 판매액 DESC ;
- 나. SELECT 지점명, 판매액 FROM 판매실적 ORDER BY 판매액 DESC ;
- 다. SELECT 지점명, 판매액 FROM 판매실적 WHERE 도시='서울' ASC ;
- 라. SELECT\*FROM 판매실적 WHEN 도시='서울' ORDER BY 판매액 DESC ;

13. SQL에서 명령어 짝의 연결이 부적절한 것은?

- 가. UPDATE.../ SET...
- 나. INSERT.../ INTO...
- 다. DELETE.../ FROM
- 라. CREATE VIEW.../ TO

14. 다음 질의를 SQL문으로 가장 잘 변환한 것은?

3학년 이상의 전자계산과 학생들의 이름을 검색하시오

- 가. SELECT\*FROM 학생 WHEN 학년>=3 AND 학과="전자계산"
- 나. SELECT 이름 FROM 학생 WHERE 학년>=3 OR 학과="전자계산"
- 다. SELECT\*FROM 학생 FOR 학년>=3 AND 학과="전자계산"
- 라. SELECT 이름 FROM 학생 WHERE 학년>=3 AND 학과="전자계산"

15. 회사원이라는 테이블에서 사원명을 찾을 때, 연락번호가 Null 값이 아닌 사원명을 모두 찾을 때의 SQL 질의로 옳은 것은?

- 가. SELECT 사원명 FROM 회사원 WHERE 연락번호!=NULL
- 나. SELECT 사원명 FROM 회사원 WHERE 연락번호 <> NULL
- 다. SELECT 사원명 FROM 회사원 WHERE 연락번호 IS NOT NULL
- 라. SELECT 사원명 FROM 회사원 WHERE 연락번호 DON'T NULL

[정답] 12.가 13.라 14.라 15.다

## [DB 6강-SQL, 시스템카탈로그, 뷰]

16. 다음 표와 같은 두 테이블에서 성별이 “여”인 사원의 “성명”, “나이”, “직책”을 구하는 SQL문은?

테이블명 : 사원

| 사번 | 성명  | 나이 | 직책 |
|----|-----|----|----|
| 11 | 홍길동 | 35 | 과장 |
| 32 | 안중근 | 44 | 부장 |
| 41 | 강동표 | 37 | 과장 |
| 31 | 송운아 | 24 | 사원 |
| 45 | 이중건 | 26 | 사원 |
| 13 | 김순자 | 33 | 계장 |

테이블명 : 연락처

| 사번 | 성별 | 연락처      |
|----|----|----------|
| 11 | 남  | 111-111  |
| 32 | 남  | 222-2222 |
| 41 | 남  | 333-3333 |
| 31 | 여  | 444-4444 |

- 가. SELECT \* FROM 사원, 연락처 WHERE 성별=“여”  
 나. SELECT 성명, 나이, 직책 FROM 사원, 연락처 WHERE 성별=“여”  
 다. SELECT 성명, 나이, 직책 FROM 사원, 연락처 WHERE 연락처.성별=“여”  
 라. SELECT 성명, 나이, 직책 FROM 사원, 연락처 WHERE 연락처.성별=“여” AND 사원.사번=연락처.사번

17. 관계 데이터베이스의 테이블 지점정보(지점코드, 소속도시, 매출액)에 대해 다음과 같은 SQL문이 실행되었다. 그 결과에 대한 설명으로 부적합한 것은?

```
SELECT 소속도시, AVG(매출액)
FROM 지점정보 WHERE 매출액 > 1000
GROUP BY 소속도시 HAVING COUNT(*) >= 3;
```

- 가. WHERE절의 조건에 의해 해당 도시의 지점들의 매출액 평균이 1,000 이하인 경우는 출력에서 제외된다.  
 나. 지점이 3군데 이상 있는 도시에 대해 각 도시별로 그 도시에 있는 매출액 1,000 초과인 지점들의 평균 매출액을 구하는 질의이다.  
 다. SELECT절의 “AVG(매출액)”을 “MAX(매출액)”으로 변경하면 각 도시별로 가장 높은 매출을 올린 지점의 매출액을 구할 수 있다.  
 라. HAVING절에서 “COUNT(\*) >= 3”을 “SUM(매출액) >= 5,000”으로 변경하면 어느 한 도시의 지점들의 매출액 합이 5,000 이상인 경우만 그 도시 지점들의 매출액 평균을 구할 수 있다.

[정답] 16.라 17.가,나

## [DB 6강-SQL, 시스템카탈로그, 뷰]

18. 다음 두 테이블 R과 S에 대한 아래 SQL문의 실행 결과로 옳은 것은?

```
SELECT A FROM R UNION SELECT A FROM S;
```

R

| A | B |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |

S

| A | C |
|---|---|
| 2 | X |
| 3 | Y |
| 4 | Z |

- 가. 1, 2, 3  
 나. 2, 3, 4  
 다. 2, 3  
 라. 1, 2, 3, 4

19. SQL문에서 STUDENT(SNO, SNAME, YEAR, DEPT) 테이블에 “학번 600, 성명 홍길동, 학년 2학년”인 학생 튜플을 삽입하는 명령으로 옳은 것은?(단, SNO는 학번, SNAME은 성명, YEAR는 학년, DEPT는 학생, 교수 구분 필드임)

- 가. INSERT STUDENT INTO VALUES(600, '홍길동', 2)  
 나. INSERT FROM STUDENT VALUES(600, '홍길동', 2)  
 다. INSERT INTO STUDENT(SNO, SNAME, YEAR) VALUES(600, '홍길동', 2)  
 라. INSERT TO STUDENT(SNO, SNAME, YEAR) VALUES(600, '홍길동', 2)

20. SQL 명령어로 수행된 결과를 실제 물리적 디스크로 저장하는 SQL 명령은?

- 가. ROLLBACK      나. COMMIT  
 다. GRANT          라. REVOKE

21. STUDENT 테이블을 생성한 후, GENDER 필드가 누락되어 이를 추가하려고 한다. 이에 적합한 SQL 명령어는?

- 가. CREATE          나. ALTER  
 다. ADD            라. MODIFY

22. SQL의 뷰(View)에 대한 장점으로 부적합한 것은?

- 가. 논리적 데이터 독립성을 제공한다.  
 나. 접근 제어를 통한 보안을 제공한다.  
 다. 뷰 정의의 변경이 용이하다.  
 라. 사용자의 데이터 관리를 간단하게 해준다.

[정답] 18.라 19.다 20.나 21.나 22.다

## [DB 6강-SQL, 시스템카탈로그, 뷰]

23. 데이터베이스 관리자가 기본 테이블에서 임의로 유도하여 만드는 테이블로서 사용자에게 접근이 허용된 자료만을 제한적으로 보여주기 위한 테이블을 무엇이라 하는가?

- 가. 임시 테이블(Temporary Table)
- 나. 뷰 테이블(View Table)
- 다. 색인 테이블(Index Table)
- 라. 기본 테이블(Base Table)

24. 뷰(View)의 이점으로 거리가 먼 것은?

- 가. 물리적인 테이블로 관리가 편하다.
- 나. 여러 사용자의 상이한 응용이나 요구를 지원해 준다.
- 다. 사용자의 데이터 관리를 간단하게 해준다.
- 라. 숨겨진 데이터를 위한 자동 보안이 제공된다.

25. 뷰(View)에 대한 설명 중 가장 거리가 먼 것은?

- 가. 뷰는 원칙적으로 하나 이상의 기본 테이블로부터 유도된 이름을 가진 가상 테이블을 말한다.
- 나. 기본 테이블은 물리적으로 구현되어 데이터가 실제로 저장되지만 뷰는 물리적으로 구현되어 있지 않다.
- 다. 뷰는 근본적으로 기본 테이블로부터 유도되지만 일단 정의된 뷰가 또 다른 뷰의 정의에 기초가 될 수도 있다.
- 라. 뷰의 정의만 시스템 내에 저장하였다가 필요시 실행 시간에 테이블을 구축하므로 시스템 검색에 있어서 뷰와 기본 테이블 사이에 약간의 차이가 있다.

26. 기본 테이블 R을 이용하여 뷰 V1을 정의하고, 뷰 V1을 이용하여 다시 뷰 V2가 정의되었다. 그리고 기본 테이블 R과 뷰 V2를 조인하여 뷰 V3를 정의하였다. 이때 다음과 같은 SQL문이 실행되면 어떤 결과가 발생하는지 올바르게 설명한 것은?

DROP VIEW V1 RESTRICT;

- 가. V1만 삭제된다.
- 나. R, V1, V2, V3 모두 삭제된다.
- 다. V1, V2, V3만 삭제된다.
- 라. 하나도 삭제되지 않는다.

27. 시스템 자신이 필요로 하는 여러 가지 객체에 관한 정보를 포함하고 있는 시스템 데이터베이스로서, 포함하고 있는 객체로는 테이블, 데이터베이스, 뷰, 접근 권한 등이 있는 것은?

- 가. 인덱스(Index)
- 나. 카탈로그(Catalog)
- 다. QBE(Query By Example)
- 라. SQL(Structure Query Language)

[정답] 23.나 24.가 25.라 26.라 27.나

## [DB 6강-SQL, 시스템카탈로그, 뷰]

28. 시스템 카탈로그에 대한 설명으로 부적합한 것은?

- 가. 데이터베이스 시스템에 따라 상이한 구조를 가진다.
- 나. 사용자도 SQL을 이용하여 검색할 수 있다.
- 다. 데이터베이스에 대한 통계 정보가 저장될 수 있다.
- 라. 사용자 데이터베이스이다.

29. 시스템 카탈로그에 대한 설명으로 옳지 않은 것은?

- 가. 시스템 자신이 필요로 하는 여러 가지 객체에 대한 정보를 포함한 시스템 데이터베이스이다.
- 나. 객체들로서는 기본 테이블, 뷰, 인덱스, 데이터베이스, 패키지, 접근 권한 등이 있다.
- 다. 카탈로그 자체도 시스템 테이블로 구성되어 있어 일반 이용자도 SQL을 이용하여 내용을 검색해 볼 수 있다.
- 라. 모든 데이터베이스 시스템에서 요구하는 정보는 동일하므로 데이터베이스 시스템의 종류에 관계없이 동일한 구조로 필요한 정보를 제공한다.

30. 시스템 카탈로그에 대한 설명으로 옳지 않은 것은?

- 가. 사용자가 시스템 카탈로그를 직접 갱신할 수 있다.
- 나. 일반 질의어를 이용해 그 내용을 검색할 수 있다.
- 다. DBMS가 스스로 생성하고, 유지하는 데이터베이스 내의 특별한 테이블의 집합체이다.
- 라. 데이터베이스 스키마에 대한 정보를 제공한다.

31. 데이터 사전(data dictionary)에 대한 설명으로 부적합한 것은?

- 가. 여러가지 스키마와 이들 속에 포함된 사상들에 관한 정보도 컴파일 되어 저장된다.
- 나. 데이터베이스를 실제로 접근하는데 필요한 정보를 유지, 관리하며 시스템만이 접근한다.
- 다. 사전 자체도 하나의 데이터베이스로 간주되며, 시스템 카탈로그(system catalog)라고도 한다.
- 라. 데이터베이스가 취급하는 모든 데이터 객체들에 대한 정의나 명세에 관한 정보를 관리 유지한다.

[정답] 28.라 29.라 30.가 31.나

## [DB 6강-SQL, 시스템카탈로그, 뷰]

32. 뷰(VIEW)에 관한 설명으로 옳지 않은 것은?

- 가. 뷰는 가상 테이블이므로 물리적으로 구현되어 있지 않다.
- 나. 하나의 뷰를 제거하면 그 뷰를 기초로 정의된 다른 뷰는 제거되지 않는다.
- 다. 필요한 데이터만 뷰로 정의해서 처리할 수 있기 때문에 관리가 용이하다.
- 라. SQL에서 뷰를 생성할 때 CREATE 문을 사용한다.

33. 뷰(view)에 대한 설명으로 옳지 않은 것은?

- 가. 데이터베이스 일부만 선택적으로 보여주므로 데이터베이스의 접근을 제한할 수 있다.
- 나. 복잡한 검색을 사용자는 간단하게 할 수 있다.
- 다. 사용자에게 데이터의 독립성을 제공할 수 있다.
- 라. 뷰는 별도의 디스크 공간을 차지하여 생성되는 실제적 테이블이다.

34. 다음 SQL 문에서 DISTINCT의 의미는?

SELECT DISTINCT DEPT FROM STUDENT ; \*

- 가. 검색결과에서 레코드의 중복을 제거하라.
- 나. 모든 레코드를 검색하라.
- 다. 검색결과를 순서대로 정렬하라.
- 라. DEPT의 처음 레코드만 검색하라.

35. 다음 SQL문 중에서 구문적 오류가 있는 것은?

- 가. DELETE FROM STUDENT, ENROL WHERE SNO = 100;
- 나. INSERT INTO STUDENT(SNO, SNAME, YEAR) VALUES (100, "홍길동", 4);
- 다. INSERT INTO COMPUTER(SNO, SNAME, YEAR) SELECT SNO, SNAME, YEAR FROM STUDENT WHERE DEPT="CE" ;
- 라. UPDATE STUDENT SET DEPT = (SELECT DEPT FROM COURSE WHERE CNO="C123") WHERE YEAR = 4;

[정답] 32.나 33.라 34.가 35.가

## [DB 6강-SQL, 시스템카탈로그, 뷰]

17. 관계 데이터베이스의 테이블 지정정보(지점코드, 소속도시, 매출액)에 대해 다음과 같은 SQL문이 실행되었다. 그 결과에 대한 설명으로 부적합한 것은?

```
SELECT 소속도시, AVG(매출액)
FROM 지정정보 WHERE 매출액 >1000
GROUP BY 소속도시 HAVING COUNT(*)>=3;
```

- 가. WHERE절의 조건에 의해 해당 도시의 지점들의 매출액 평균이 1,000 이하인 경우는 출력에서 제외된다.
- 나. 지점이 3군데 이상 있는 도시에 대해 각 도시별로 그 도시에 있는 매출액 1,000 초과인 지점들의 평균 매출액을 구하는 질의이다.
- 다. SELECT절의 "AVG(매출액)"을 "MAX(매출액)"으로 변경하면 각 도시별로 가장 높은 매출을 올린 지점의 매출액을 구할 수 있다.
- 라. HAVING절에서 "COUNT(\*)>=3"을 "SUM(매출액)>=5,000"으로 변경하면 어느 한 도시의 지점들의 매출액 합이 5,000 이상인 경우만 그 도시 지점들의 매출액 평균을 구할 수 있다.

[지점정보]

| 지점코드    | 소속도시 | 매출액  |
|---------|------|------|
| A000001 | A    | 1000 |
| A000002 | B    | 2000 |
| A000003 | B    | 3000 |
| A000004 | B    | 4000 |
| A000005 | A    | 5000 |
| A000006 | A    | 3000 |

- 1) WHERE절 처리
  - 전체에서 매출액이 1000 초과만 해당됩니다.
  - 지점코드 A000002,A000003,A000004,A000005,A000006
- 2) GROUP By절 처리
  - 해당 레코드에서 소속도시를 기준으로 그룹화
- 3) HAVING절 처리
  - 그룹의 건수가 3개 이상인 소속도시는 B만 존재
- 4) SELECT 처리
  - B 와 매출액 평균(3000)을 SELECT 합니다.

그러므로

가번은 매출액 평균을 매출액으로 수정  
 나번은 매출액이 1000 초과하는 지점이 3개 이상인 도시의 소속도시별 매출액 평균을 구하라 입니다.  
 → 정답은 가,나 입니다.



## [DB 6강-SQL, 시스템카탈로그, 뷰]

[기-08년9월][산-08년3월]

36. 시스템 카탈로그에 대한 설명으로 옳지 않은 것은?

- 가. 기본 테이블, 뷰, 인덱스, 패키지, 접근 권한 등의 정보를 저장한다.
- 나. 시스템 테이블로 구성되어 있어 일반 사용자는 내용을 검색할 수 없다.
- 다. 시스템 자신이 필요로 하는 스키마 및 여러 가지 객체에 대한 정보를 포함하고 있는 시스템 데이터베이스이다.
- 라. 자료 사전(Data Dictionary)이라고도 한다.

[산-08년5월][기-08년3월]

37. 뷰(View)에 대한 설명으로 옳지 않은 것은?

- 가. 뷰는 독자적인 인덱스를 가질 수 없다.
- 나. 뷰의 정의를 변경할 수 없다.
- 다. 뷰로 구성된 내용에 대한 삽입, 갱신, 삭제 연산에는 제약이 따른다.
- 라. 뷰가 정의된 기본 테이블이 삭제되더라도 뷰는 자동적으로 삭제되지 않는다.

[기-08년3월]

38. 데이터베이스 언어 중 DDL의 기능이 아닌 것은?

- 가. 논리적, 물리적 데이터 구조의 정의
- 나. 데이터 회복과 병행 수행 제어
- 다. 논리적 데이터 구조와 물리적 데이터 구조의 사상 정의
- 라. 데이터베이스 정의 및 수정

[기-08년3월]

39. 시스템 카탈로그에 대한 설명으로 옳은 것은?

- 가. 메타 데이터를 갖고 있는 시스템 데이터베이스이다.
- 나. 일반 사용자도 제한 없이 시스템 카탈로그의 내용을 직접 갱신할 수 있다.
- 다. 시스템 카탈로그는 사용자의 테이블당 한 개씩 만들어진다.
- 라. 시스템 카탈로그는 DBA가 생성한다.

[산-08년9월]

40. 뷰(View)의 설명으로 거리가 먼 것은?

- 가. 뷰는 저장장치 내에 물리적으로 존재하지 않지만, 사용자에게는 있는 것처럼 간주된다.
- 나. 뷰를 통하여 데이터를 접근하게 되면 뷰에 나타나지 않는 데이터는 안전하게 보호할 수 있다.
- 다. 필요한 데이터만 뷰로 정의해서 처리할 수 있기 때문에 관리가 용이해진다.
- 라. 삽입, 삭제 연산에 아무런 제한이 없으므로 사용자가 뷰를 다루기가 편하다.

[정답] 36.나 37.라 38.나 39.가 40.라

## [DB 7강-내장 SQL, 고급데이터베이스]

[출제빈도 '상']

### 1. 내장 SQL(Embedded-SQL) ★★☆☆☆

: 호스트 언어(C, C++, 비주얼베이직 등)에 삽입된 SQL → 목적 : 일괄처리, 동일 업무 반복 시

EXEC SQL BEGIN DECLARE SECTION;

char 물품번호;

char 이름;

int 단가;

EXEC SQL END DECLARE SECTION;

물품번호='A001';

EXEC SQL SELECT 이름, 단가

INTO :이름, :단가

FROM Product

WHERE 물품번호 = :물품번호;

END EXEC

\* 변수 선언

- 테이블의 속성을 변수로 선언
- 변수 타입은 속성 타입과 같아야 함

\* 삽입되는 SQL 문

- INTO 절에는 검색 결과가 저장될 변수가 지정됨.

:이름 → SQL 내 속성명과 구분하기 위해

호스트 변수를 이용할 경우 : 을 붙인다.

내장 SQL 문장 끝은 호스트 언어의 종류에 따라 종료를 표시하는 방법이 다르다. (일반적으로 세미콜론 ;)

## [DB 7강-내장 SQL, 고급데이터베이스]

### 2. SQL CODE (SQL 상태) ★☆☆☆☆

: 내장 SQL에서 SQLCODE(정수 타입으로 선언해야 한다.)라는 변수 사용시는 SQL 명령문이 실행되고 나서 이 변수에 0 값이 SET되게 되면 **성공적** 실행 상태를 나타내는 것이고, 100이 SET되게 되면 **실패**, 양수가 SET되게 되면 **경고**, 음수가 SET되게 되면 **에러**를 나타내게 된다.

## [DB 7강-내장 SQL, 고급데이터베이스]

### 1. 트랜잭션(TRANSACTION) 정의

: 데이터베이스의 상태를 변화시키는 논리적 연산의 집합

### 2. 트랜잭션 특징 ★★★★★

#### 1) 원자성(Atomicity)

- 모두 반영되거나 아니면 전혀 반영되지 않아야 된다. (부분 실행 안됨)

#### 2) 일관성(Consistency)

- 트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있게 DB 상태로 변환

- 시스템이 가지고 있는 고정요소는 트랜잭션 수행 전과 트랜잭션 수행 완료 후에 같아야 한다

#### 3) 독립성, 격리성(isolation)

- 둘 이상의 트랜잭션이 동시에 병행 실행되고 있을 때 또 다른 하나의 트랜잭션의 연산이 끼어들 수 없다.

#### 4) 영속성, 지속성(Durability)

- 트랜잭션의 결과는 영구적으로 반영

\* 무결성, 정확성, 투명성, 병행성, INDEPENDENCY (X)

### 3. 연산의 종류 ★★☆☆☆

1) **Commit** : 한 작업의 논리적 단위가 성공적으로 끝났고, 데이터베이스가 다시 일관된 상태에 있으며 이 트랜잭션이 행한 갱신 연산이 완료된 것을 트랜잭션 관리자에게 알려주는 연산

2) **Rollback** : 트랜잭션의 실행이 실패하였음을 알리는 연산자로 트랜잭션이 수행한 결과를 원래의 상태로 원상 복구시키는 연산

## [DB 7강-내장 SQL, 고급데이터베이스]

### 1. 회복 정의

: 트랜잭션 수행도중 장애가 발생하여 데이터베이스가 손상 입었기에 손상되기 이전 상태로 복구하는 작업

### 2. 장애의 유형 ★☆☆☆☆

1) 트랜잭션 장애 2) 시스템 장애 3) 미디어 장애

→ 트랜잭션 장애가 데이터베이스에 손상을 줄 가능성이 가장 적은 장애

## [DB 7강-내장 SQL, 고급데이터베이스]

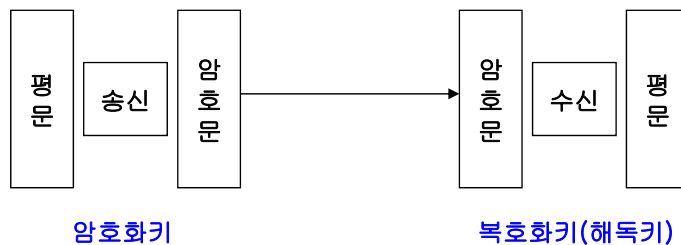
### 1. 보안, 암호화 ★★☆☆☆

1) 개인키(Private) 암호화 알고리즘 : DES (암호화키 = 복호화키)

- 동일한 키를 이용하는 방식 → 보안 수준이 낮음 → 알고리즘이 단순하고, 빠르다.

2) 공개키(Public) 암호화 알고리즘 : RSA (암호화키 <> 복호화키)

- 서로 다른 키를 사용하는 비대칭 암호화 방식 → 보안 수준이 높음 → 속도가 느리고 알고리즘 복잡. 파일 크기도 크다.



\* 보안과 무결성

- 무결성은 권한이 있는 사용자로부터 데이터베이스를 보호하는 것이고,

- 보안은 권한이 없는 사용자로부터 데이터베이스를 보호하는 것이다.

\* 보안을 위한 사용자들의 권한부여는 관리자의 정책에 의해 결정되어 진다.



## [DB 7강-내장 SQL, 고급데이터베이스]

### 1. 병행제어 정의

: 동시에 여러 개 수행할 때, 데이터베이스 일관성 유지를 위해 트랜잭션 간의 상호 작용을 제어

### 2. 병행제어 목적 ★☆☆☆☆

- 1) 데이터베이스 공유 최대화
- 2) 시스템 활용도 최대화
- 3) 데이터베이스 일관성 유지
- 4) 사용자에게 대한 응답시간 최소화

### 3. 병행제어 기법 ★★★☆☆

- 1) 로킹 (Locking)
  - 하나의 트랜잭션이 데이터를 액세스하는 동안 다른 트랜잭션이 그 데이터 항목을 액세스할 수 없도록 하는 방법
- 2) 로킹 단위
  - 병행제어에서 한꺼번에 로킹 할 수 있는 단위
  - 로킹 단위가 크면 → 로크수가 작다. → 관리가 수월해지고 병행성 수준↓
  - 로킹 단위가 작으면 → 로크수가 커진다. → 관리가 복잡해지고 병행성 수준↑

### 4. 병행수행 허용시 발생하는 문제점 ★☆☆☆☆

- 1) 갱신분실 : 2개 이상의 트랜잭션이 같은 자료를 갱신할 때 일부가 없어지는 현상
- 2) 비완료 의존성 : 하나의 트랜잭션이 실패한 후 다른 트랜잭션이 실패한 갱신 결과를 참조하는 현상
- 3) 불일치 : 원치 않는 자료를 이용하는 현상

## [DB 7강-내장 SQL, 고급데이터베이스]

### 1. 분산 데이터베이스 정의

: 컴퓨터 네트워크 상에 물리적으로 분산된 데이터베이스를 논리적으로는 1개로 인식하는 기법

### 2. 분산 데이터베이스 4대 목표 ★★★☆☆

- 1) 위치 투명성
  - 사용자가 물리적으로 저장되어 있는 곳을 알 필요 없이 논리적인 입장에서 데이터가 모두 자신의 사이트에 있는 것처럼 처리
- 2) 중복(복제) 투명성
  - 트랜잭션이 데이터의 중복 개수나 중복 사실을 모르고도 데이터 처리가 가능
- 3) 병행 투명성
  - 분산 데이터베이스와 관련된 다수의 트랜잭션들이 동시에 실행 되더라도 그 트랜잭션의 결과는 영향을 안 받음
- 4) 장애 투명성
  - 트랜잭션, DBMS, 네트워크, 컴퓨터 장애에도 불구하고 트랜잭션을 정확하게 처리함

### 3. 분산 데이터베이스의 특징 ★★★★★

- 1) 자료 공유 용이
- 2) 시스템 성능 향상
- 3) 점증적 시스템 용량 확장 용이
- 4) 설계가 어렵고, 소프트웨어 개발 비용 증가
- 5) 오류 발생 가능성 높음

## [DB 7강-내장 SQL, 고급데이터베이스]

1. 내장 SQL에 대한 설명으로 옳지 않은 것은?

- 가. 내장 SQL 실행문은 호스트 실행문이 나타날 수 있는 곳이면, 어디에서나 사용 가능하다.
- 나. SQL문에 사용되는 호스트 변수는 콜론(:)을 앞에 붙인다.
- 다. 응용 프로그램에서 삽입 SQL문은 'EXEC SQL'을 앞에 붙여 다른 호스트 명령문과 구별한다.
- 라. 내장 SQL문의 호스트 변수의 데이터 타입은 이에 대응하는 데이터베이스 필드의 SQL 데이터 타입과 일치하지 않아도 된다.

2. 삽입 SQL(Embedded SQL)에 대한 설명으로 옳지 않은 것은?

- 가. 응용 프로그램에 삽입되어 사용되는 SQL이다.
- 나. SQL 문장의 식별자로서 EXEC SQL을 앞에 기술한다.
- 다. 호스트 변수와 데이터베이스 필드의 이름은 같아도 무방하다.
- 라. 호스트 언어의 변수는 SQL 변수와 구별하기 위하여 앞에 % 기호를 붙인다.

3. 내장(Embedded) SQL에 대한 설명으로 옳지 않은 것은?

- 가. 내장 SQL문은 EXEC SQL이 앞부분에 위치한다.
- 나. SQL에 사용되는 호스트 변수는 콜론(:)을 앞에 붙인다.
- 다. SQL CODE의 값이 음수인 경우 경고를 의미한다.
- 라. 내장 SQL 프로그램은 컴파일보다 우선하는 전처리에 의해 처리된다.

4. 트랜잭션에 대한 설명으로 옳지 않은 것은?

- 가. 트랜잭션은 작업의 논리적 단위이다.
- 나. 하나의 트랜잭션은 Commit되거나 Rollback되어야 한다.
- 다. 트랜잭션은 일반적으로 회복의 단위가 된다.
- 라. 구조점(Savepoint)은 트랜잭션당 한 번만 지정할 수 있다.

5. 트랜잭션(Transaction)이 가져야 할 특성에 해당하지 않는 것은?

- 가. 원자성(Atomicity)      나. 투명성(Transparency)
- 다. 일관성(Consistency)      라. 격리성(Isolation)

[정답] 1.라 2.라 3.다 4.라 5.나

## [DB 7강-내장 SQL, 고급데이터베이스]

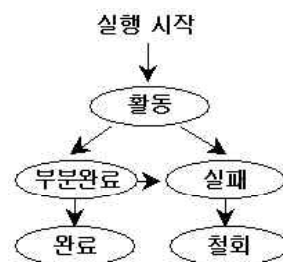
6. 트랜잭션(Transaction)의 특성에 대한 설명으로 옳지 않은 것은?

- 가. 원자성(Atomicity)은 트랜잭션의 일부만 수행된 상태로 종료될 수 있다는 특성을 의미한다.
- 나. 일관성(Consistency)은 시스템의 고정 요소는 트랜잭션 수행 전과 수행 완료 후에 같아야 한다는 특성을 의미한다.
- 다. 고립성(Isolation)은 트랜잭션이 실행될 때마다 다른 트랜잭션의 간섭을 받지 않아야 한다는 성질을 의미한다.
- 라. 지속성(Durability)은 트랜잭션의 완료 결과가 데이터베이스에 영구히 기억되는 성질을 의미한다.

7. 한 작업의 논리적 단위가 성공적으로 끝났고, 데이터베이스가 다시 일관된 상태에 있으며, 이 트랜잭션이 행한 갱신 연산이 완료 된 것을 트랜잭션 관리자에게 알려주는 연산은?

- 가. Rollback 연산
- 나. Log 연산
- 다. Commit 연산
- 라. Backup 연산

8. 다음 그림은 트랜잭션 상태를 나타내고 있다. 각 상태에 대한 설명으로 옳지 않은 것은?



- 가. 활동(Active) - 초기상태로 트랜잭션이 Begin\_Trans에서부터 실행을 시작하였거나 실행중인 상태
- 나. 부분 완료(Partially Committed) - 트랜잭션이 명령문 중 일부를 실행한 직후의 상태
- 다. 철회(Aborted) - 트랜잭션이 실행에 실패하여 Roll-back 연산을 수행한 상태
- 라. 완료(Committed) - 트랜잭션이 실행을 성공적으로 완료 연산을 수행한 상태

9. A 은행에서 B 라는 사람이 고객 인증 절차를 거쳐 잔액을 조회한 후, 타인에게 송금하는 도중에 장애가 발생하였을 경우 문제가 발생한다. 이러한 경우의 부작용을 방지할 수 있는 트랜잭션의 특성은?

- 가. 일관성(Consistency)      나. 고립성(Isolation)
- 다. 지속성(Duration)      라. 원자성(Atomicity)

[정답] 6.가 7.다 8.나 9.라

## [DB 7강-내장 SQL, 고급데이터베이스]

10. 트랜잭션의 병행제어 목적이 아닌 것은?

- 가. 데이터베이스의 공유 최대화
- 나. 시스템의 활용도 최대화
- 다. 데이터베이스의 일관성 최소화
- 라. 사용자에 대한 응답시간 최소화

11. 데이터베이스 운영 시 데이터베이스에 손상이 가해져 회복(Recovery) 작업을 해야 하는 경우가 발생할 수 있는데, 이런 경우 데이터베이스에 손상을 줄 가능성이 가장 적은 장애는?

- 가. 트랜잭션 장애(Transaction Failure)
- 나. 시스템 장애(System Failure)
- 다. 미디어 장애(Media Failure)
- 라. 네트워크 장애(Network Failure)

12. 하나의 트랜잭션이 데이터를 액세스하는 동안 다른 트랜잭션이 그 데이터 항목을 액세스할 수 없도록 하는 방법을 무엇이라고 하는가?

- 가. Normalization(정규화)
- 나. Locking(로킹)
- 다. Logging(사용 흔적의 일지화)
- 라. Fire Wall(방화벽)

[정답] 10.다 11.가 12.나 13.라 14.가  
15.라 16.나

13. 병행처리를 수행할 때 로킹 방법에 의해 해결될 수 있는 것이 아닌 것은?

- 가. 갱신 분실 문제      나. 비완료 의존성 문제
- 다. 불일치 문제      라. 교착 상태 문제

14. 로킹(Locking) 단위에 대한 설명으로 옳은 것은?

- 가. 로킹 단위가 크면 병행성 수준이 낮아진다.
- 나. 로킹 단위가 크면 병행제어 기법은 복잡해진다.
- 다. 로킹 단위가 작으면 로크(Lock)의 수가 적어진다.
- 라. 로킹 단위가 크면 로크(Lock)의 수가 많아진다.

15. 데이터 무결성과 가장 관계가 깊은 것은?

- 가. 데이터의 안전성      나. 데이터의 공유성
- 다. 데이터의 중복성      라. 데이터의 정확성

16. 암호화 기법 중 암호화 알고리즘과 암호화 키는 공개해서 누구든지 평문을 암호문으로 만들 수 있지만, 해독 알고리즘과 해독키는 비밀로 유지하는 기법을 무엇이라 하는가?

- 가. DES(Data Encryption Standard) 기법
- 나. 공중키(Public Key) 암호화 기법
- 다. 대체(Substitution) 암호화 기법
- 라. 전치(Transposed) 암호화 기법

## [DB 7강-내장 SQL, 고급데이터베이스]

17. 분산 데이터베이스의 불법적인 접근을 차단하기 위하여 데이터 암호화가 필요하다. DES 알고리즘에서는 평문을 (①) 비트로 블록화를 하고, 실제키의 길이는 (②) 비트를 이용한다. 괄호의 내용으로 옳은 것은?

- 가. ①64    ②56      나. ①64    ②32
- 다. ①32    ②16      라. ①32    ②8

18. 다음 SQL에서 데이터베이스 사용자에게 부여하는 권한에 대한 설명으로 옳은 것은?

GRANT RESOURCE, CONNECT TO 사용자;

- 가. 사용자에게 데이터베이스 객체를 생성하고 데이터베이스에 접속할 권한을 준다.
- 나. 사용자에게 시스템 자원을 사용할 수 있게 하고 데이터베이스에 접속할 권한을 준다.
- 다. 사용자에게 시스템 자원을 사용할 수 있게 하고 다른 사용자의 스키마에 들어갈 수 있는 권한을 준다.
- 라. 사용자에게 데이터베이스 객체를 다른 사용자의 스키마에 들어갈 수 있는 권한을 준다.

19. 분산 데이터베이스의 장점으로 거리가 먼 것은?

- 가. 데이터베이스 관련 소프트웨어 개발 비용 감소
- 나. 신뢰성(Reliability)과 가용성(Availability) 향상
- 다. 질의 처리(Query Processing) 시간의 단축
- 라. 데이터의 공유성 향상

20. 분산 데이터베이스의 장점으로 거리가 먼 것은?

- 가. 지역 자치성이 높다.
- 나. 효율성과 융통성이 높다.
- 다. 점증적 시스템 용량 확장이 용이하다.
- 라. 소프트웨어 개발 비용이 저렴하다.

21. 분산 데이터베이스 시스템이 사용자에게 제공하는 4가지 유형의 투명성(Transparency)에 속하지 않는 것은?

- 가. 위치 투명성      나. 복제 투명성
- 다. 수행 투명성      라. 병행 투명성

[정답] 17.가 18.가 19.가 20.라 21.다

## [DB 7강-내장 SQL, 고급데이터베이스]

22. 트랜잭션이 갖는 특성으로 틀린 것은?

- 가. 원자성(atomicity)    나. 일관성(consistency)  
다. 독립성(isolation)    라. 유지보수성(maintenance)

23. 트랜잭션이 가져야 되는 특성과 거리가 먼 것은?

- 가. 원자성(atomicity)  
나. 일관성(consistency)  
다. 독립성(independency)  
라. 영속성(durability)

[정답] 22.라 23.다

## [DB 8강]-자료 구조(선형, 비선형구조)

[출제빈도 '상']

### 1. 자료구조 분류

: 자료를 기억장치 내에 저장하는 방법

- 1) 선형 구조 : 순차 리스트 (스택, 큐, 덱, 배열), 연결 리스트
- 2) 비선형 구조 : Tree, Graph

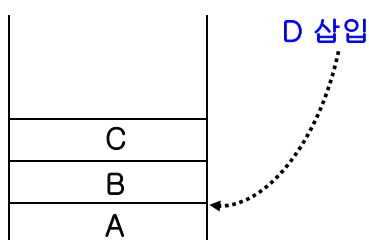
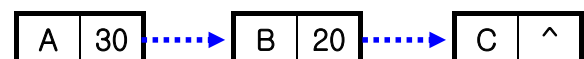
### 2. 선형 구조 ★☆☆☆☆

1) **순차 리스트** (선형, Sequential List) : **연속적**인 기억장소에 저장

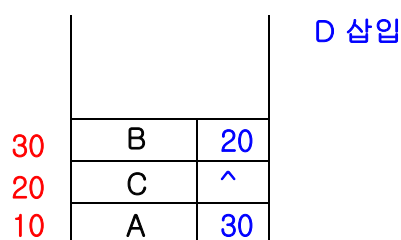
- 아파트를 계단으로 연속적으로 이동
- 특징 : 구조 간단, 기억장소 이용 효율이 높음, 삽입/삭제 어려움, 연결리스트에 비해 검색 빠름.

2) **연결 리스트** (Linked List) : **비연속적**으로 저장

- 아파트를 엘리베이터로 비연속적으로 이동 (포인터)
- 특징 : 기억장소 이용 효율이 낮음, 삽입/삭제 용이, 순차 리스트에 비해 검색 느림.



[순차 리스트]

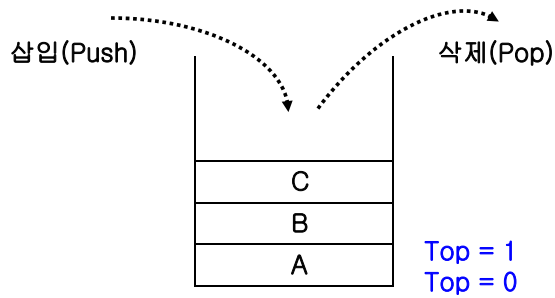


[연결 리스트]

## [DB 8강]-자료 구조(선형, 비선형구조)

### 3. 선형구조 > 순차리스트 > 스택 (Stack) ★★★★★

1) 삽입/삭제가 한 쪽에서 이루어지는 데이터 구조 (LIFO : Last In First Out)



- \* Top Point : 가장 최근에 삽입된 자료 또는 가장 먼저 삭제될 자료를 가리키는 스택 포인터
- 삽입 : Top 값 증가
- 삭제 : Top 값 감소

#### 2) 스택 응용 분야

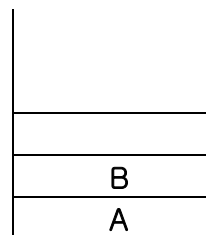
- 인터럽트의 처리
- 수식의 계산
- 서브루틴의 복귀번지 저장
- 부프로그램(sub program)의 호출 = 함수 호출의 순서제어

#### 운영체제의 작업 스케줄링(X)

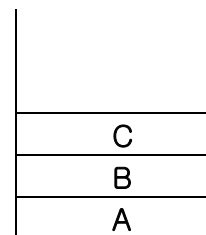
## [DB 8강]-자료 구조(선형, 비선형구조)

### 2) 삽입 알고리즘

```
Top = Top + 1
If(Top > M) Then
    Stack_overflow
Else
    Stack(Top) ← data
```



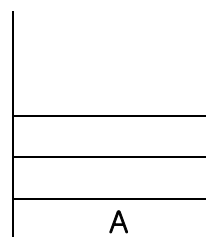
Top = 2 , M = 3



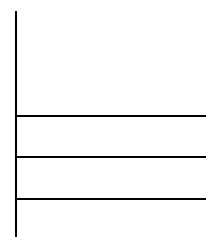
Top = 3 , M = 3

### 3) 삭제 알고리즘

```
If(Top = 0) Then
    Stack_Empty
Else
    data ← Stack(Top)
    Top = Top - 1
```



Top = 1



Top = 0

## [DB 8강]-자료 구조(선형, 비선형구조)

스택에 데이터를 A, B, C, D 순으로 저장했을 경우, 이들 데이터가 출력되는 결과로 가능한 것은 ? [정답 나]

- 가. D-B-C-A      나. C-B-D-A  
다. D-C-A-B      라. D-A-C-B

➔ 문제에서 저장했을 경우를 저장할 경우로 해석해야 이 문제를 해결할 수 있습니다.

가 : A(PUSH) → B(PUSH) → C(PUSH) → D(PUSH) → D(POP)

➔ 그 다음 B(POP) 할 수 없습니다. C(POP)만 가능하죠. C보다 B를 POP 할 수 없습니다.

나 : A(PUSH) → B(PUSH) → C(PUSH) → C(POP) → B(POP) → D(PUSH) → D(POP) → A(POP)

➔ 이해하실 내용은 꼭 A,B,C,D를 다 PUSH 후에 POP을 할 필요 없다는 것입니다.

다 : A(PUSH) → B(PUSH) → C(PUSH) → D(PUSH) → D(POP) → C(POP)

➔ 그 다음 A(POP) 할 수 없습니다.

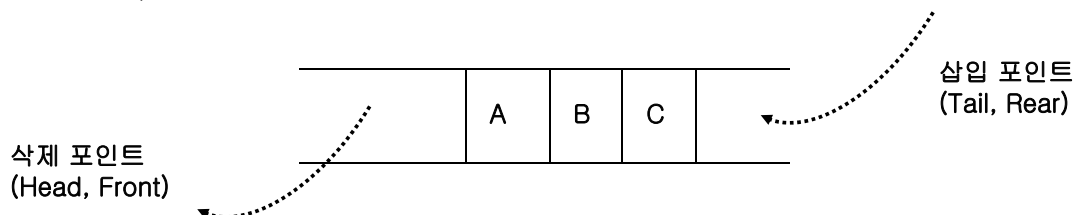
라 : A(PUSH) → B(PUSH) → C(PUSH) → D(PUSH) → D(POP)

➔ 그 다음 A(POP) 할 수 없습니다.

## [DB 8강]-자료 구조(선형, 비선형구조)

### 4. 선형구조 > 순차리스트 > 큐(Queue) ★★★☆☆

- 1) 노드의 삽입 작업은 선형 리스트의 한 쪽 끝에서, 제거 작업은 다른 쪽 끝에서 수행되는 자료 구조 (FIFO : First In First Out)

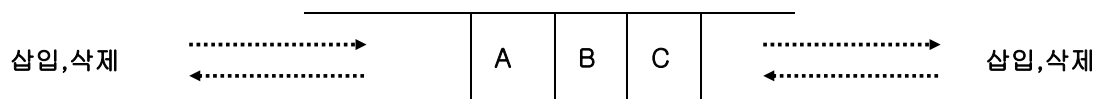


- 2) 응용 분야

- 운영체제의 작업 스케줄링, 키보드 버퍼 이용 시, 스푼(spool) 운용 시

### 5. 선형구조 > 순차리스트 > 데크(Deque) ★★★☆☆

- 1) 삽입과 삭제가 리스트의 양쪽 끝에서 발생할 수 있는 자료 구조 (double ended queue)



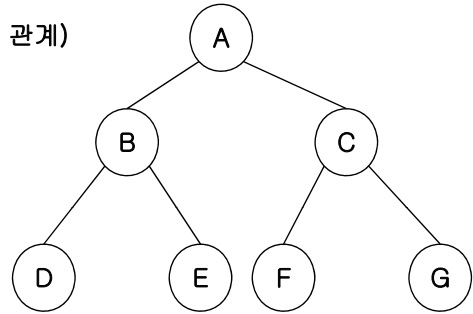
## [DB 8강]-자료 구조(선형, 비선형구조)

### 1. 비선형구조 > 트리 (Tree) ★★★★★

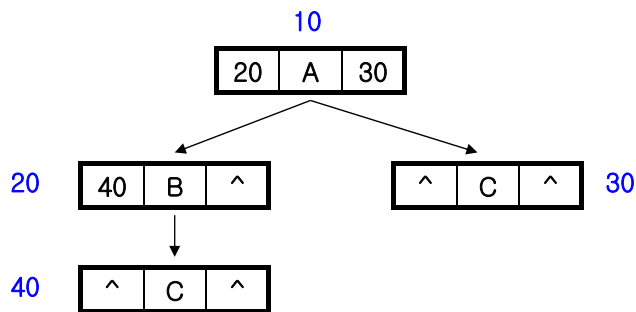
: 노드와 간선으로 구성되어 있고, 사이클이 없다. (족보, 부모 자식 관계)

1) 용어

- Node (노드)
- Root Node (근노드) : 최상위 노드
- 노드의 차수 (degree) : 어떤 노드의 서브트리 수
- 트리의 차수 : 노드의 차수 값 중 최대값
- 단말노드 (Terminal, Leaf) : 차수가 0 인 노드
- 형제노드 : 같은 부모 노드를 가지는 노드
- 레벨(Level) : 노드의 깊이



2) 특징 : 연결 리스트 구조로 표현 (포인터 이용)



## [DB 8강]-자료 구조(선형, 비선형구조)

### 2. 트리의 종류 > 이진 트리 ★☆☆☆☆

: 차수가 2 이하로 구성된 트리

- 이진 트리의 레벨 i 에서 최대 노드의 수 :  $2^i - 1$

### 3. 트리의 종류 > 스레드(threaded) 이진 트리 ★☆☆☆☆

: 기억 공간의 낭비 원인이 되는 널 링크 부분을 트리 순회 시 이용되도록 구성한 트리

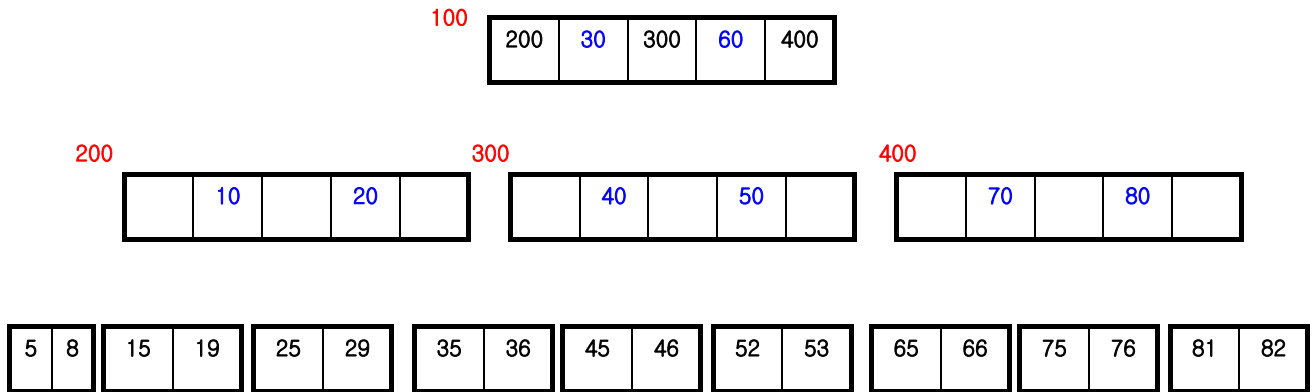
- 널 링크를 다른 노드를 가리키는 포인터로 대체한다.
- 스택의 도움 없이 트리를 순회할 수 있는 장점이 있다.
- 실제 포인터와 스레드를 구별하기가 어렵다

## [DB 8강]-자료 구조(선형, 비선형구조)

### 4. 트리의 종류 > B - 트리 ★☆☆☆☆

: 인덱스 파일에서 인덱스를 구성하는 방법 중의 하나

- 한 노드 안에 있는 키값은 오름차순을 유지한다.
- 모든 리프노드(단노드)는 같은 레벨에 있다. (균형 유지)
- 루트노드는 리프가 아닌 이상 적어도 두개의 서브트리를 갖는다.
- 탐색, 추가, 삭제는 루트로부터 시작한다.
- 인덱스 파일에서 인덱스를 구성하는 방법 중 하나다.



## [DB 8강]-자료 구조(선형, 비선형구조)

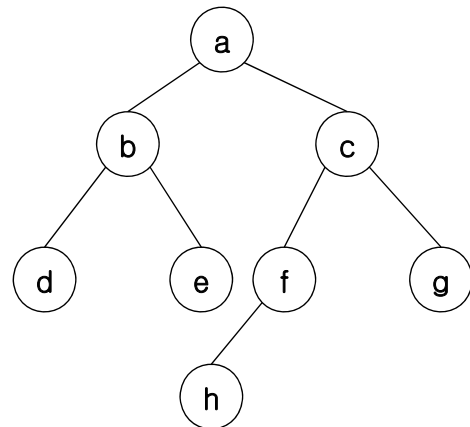
### 5. 이진 트리 운행법 ★★★★★

: 트리를 구성하는 노드들을 찾아가는 방법

- Preorder (전위) : Root → Left → Right
- Inorder (중위) : Left → Root → Right
- Postorder (후위) : Left → Right → Root

예)

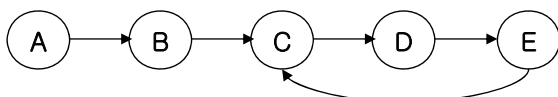
- Preorder (전위) : a, b, d, e, c, f, h, g
- Inorder (중위) : d, b, e, a, h, f, c, g
- Postorder (후위) : d, e, b, h, f, g, c, a



### 6. Graph ★☆☆☆☆

: 노드와 간선으로 구성되어 있고, 사이클이 있다.

1) 그래프를 인접행렬로 표시



|   | A | B | C | D | E |
|---|---|---|---|---|---|
| A | 0 | 1 | 0 | 0 | 0 |
| B | 0 | 0 | 1 | 0 | 0 |
| C | 0 | 0 | 0 | 1 | 0 |
| D | 0 | 0 | 0 | 0 | 1 |
| E | 0 | 0 | 1 | 0 | 0 |



## [DB 8강]-자료 구조(선형, 비선형구조)

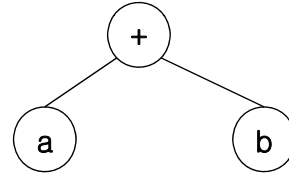
### 7. 수식 표기법 변환 ★★★★★☆

: 산술식을 계산하기 위해 기억공간에 기억시키는 방법으로 이진 트리를 많이 이용한다.

- PreFix (전위 표기법) : 연산자 → Left 피연산자 → Right 피연산자
- InFix (중위 표기법) : Left 피연산자 → 연산자 → Right 피연산자
- PostFix (후위 표기법) : Left 피연산자 → Right 피연산자 → 연산자

예)

- PreFix (전위 표기법) : + a b
- InFix (중위 표기법) : a + b
- PostFix (후위 표기법) : a b +

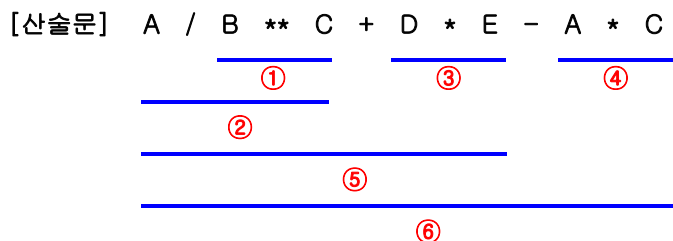


## [DB 8강]-자료 구조(선형, 비선형구조)

주의)

- 연산 우선 순위 : 괄호 → \*\* (거듭제곱) → \*, / → +, -

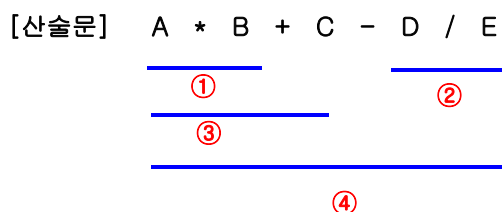
### 1) InFix → PostFix



[풀이]

- ① : BC\*\*
- ② : A / ① → ABC\*\*/
- ③ : DE\*
- ④ : AC\*
- ⑤ : ② + ③ → ABC\*\*/DE\*\*+
- ⑥ : ⑤ - ④ → ABC\*\*/DE\*\*+AC\*-

### 2) InFix → PreFix



[풀이]

- ① : \*AB
- ② : /DE
- ③ : ① + C → ++ABC
- ④ : ③ - ② → -++ABC/DE

## [DB 8강]-자료 구조(선형, 비선형구조)

주의) : 왼쪽에서 PreFix 표기인 연산자, 피연산자, 피연산자 구조를 찾는다.

### 3) PreFix → PostFix

[산술문]    -    /    \*    A    +    B    C    D    E

[풀이]

① : BC+

② : \* A ① → ABC+\*

③ : / ② D → ABC+\*D/

④ : - ③ E → ABC+\*D/E-

주의) : 왼쪽에서 PostFix 표기인 피연산자, 피연산자, 연산자 구조를 찾아서 괄호로 묶는다.

### 4) PostFix → InFix

[산술문]    A    B    C    -    /    D    E    F    +    \*    +

((A (B C -) /) (D (E F +) \*) +)

[풀이]

① : (B-C)

② : (A ① /) → (A/(B-C))

③ : (E+F)

④ : (D ③ \*) → (D\*(E+F))

⑤ : (② ④ +) → ((A/(B-C))+(D\*(E+F)))

→ 필요없는 괄호 없애기

: A/(B-C)+D\*(E+F)

## [DB 8강]-자료 구조(선형, 비선형구조)

1. 순차 리스트(Sequential List)가 아닌 것은?

가. 배열(Array)    나. 트리(Tree)  
다. 데크(Deque)    라. 스택(Stack)

2. 순차적인 선형 구조(Sequential Linear Structure)에 해당되는 자료 구조는?

가. 트리    나. 연결 리스트  
다. 그래프    라. 큐

3. 비선형 구조와 선형 구조가 옳게 짝지어진 것은?

①스택(Stack)    ②큐(Queue)    ③트리(Tree)  
④연결 리스트(Linked List)    ⑤그래프(Graph)

가. 비선형 구조 : ①,②,⑤    선형구조 : ③,④  
나. 비선형 구조 : ③,⑤    선형구조 : ①,②,④  
다. 비선형 구조 : ①,②,③    선형구조 : ④,⑤  
라. 비선형 구조 : ③    선형구조 : ①,②,④,⑤

4. 선형 리스트의 특징이 아닌 것은?

가. 가장 간단한 데이터 구조 중 하나이다.  
나. 배열과 같이 연속되는 기억장소에 저장되는 리스트를 말한다.  
다. 기억장소 효율을 나타내는 메모리 밀도가 1이다.  
라. 데이터 항목을 추가, 삭제하는 것이 용이하다.

5. 연결 리스트(Linked List)에 대한 설명으로 거리가 먼 것은?

가. 노드의 삽입이나 삭제가 쉽다.  
나. 노드들이 포인터로 연결되어 검색이 빠르다.  
다. 연결을 해주는 포인터(Pointer)를 위한 추가 공간이 필요하다.  
라. 연결 리스트 중에는 중간 노드 연결이 끊겨지면 그 다음 노드를 찾기 힘들다.

6. 회소 행렬을 링크드 리스트(Linked List)로 표현할 때 가장 큰 장점은?

가. 기억장소가 절약된다.  
나. 임의 위치 액세스(Random Access)가 가능하다.  
다. 이진 검색(Binary Search)이 가능하다.  
라. 행렬간의 연산시간을 줄일 수 있다.

[정답] 1.나 2.라 3.나 4.라 5.나 6.가

## [DB 8강]-자료 구조(선형, 비선형구조)

7. 리스트 내의 데이터 삽입, 삭제가 한 쪽 끝에서 이루어지는 데이터 구조를 무엇이라 하는가?

- 가. 스택(Stack)    나. 큐(Queue)  
다. 데크(Deque)    라. 원형 큐(Circular Queue)

8. 스택 알고리즘에서 T가 스택 포인터이고, m이 스택의 길이 일때 서브루틴 AA가 처리해야 하는 것은?

```
T←T+1
If T > m Then
    Goto AA
Else
    X(T)←Y
```

- 가. 오버플로 처리  
나. 언더플로 처리  
다. 입력 처리  
라. 출력 처리

9. 스택(Stack)의 응용 분야와 거리가 먼 것은?

- 가. 인터럽트의 처리    나. 수식의 계산  
다. 서브루틴의 복귀번지 저장  
라. 운영체제의 작업 스케줄링

10. 스택 메모리에 대한 정보의 입출력 방식은?

- 가. FIFO    나. FILO    다. LILO    라. LIFO

11. 노드의 삽입 작업은 선형 리스트의 한 쪽 끝에서, 제거 작업은 다른 쪽 끝에서 수행되는 자료 구조는?

- 가. 스택    나. 큐    다. 트리    라. 그래프

12. 운영체제의 작업 스케줄링 등에 응용되는 것으로 가장 적합한 자료 구조는?

- 가. 스택(Stack)    나. 큐(Queue)  
다. 연결리스트(Linked List)    라. 트리(Tree)

13. 자료구조 중 먼저 입력된 자료가 먼저 출력되는 형태로 헤드(Head)와 테일(Tail)을 입.출력 포인터로 사용하는 자료구조 형태는?

- 가. 스택(Stack)    나. 큐(Queue)  
다. 데크(Deque)    라. 포인터(Pointer)

[정답] 7.가 8.가 9.라 10.나,라 (복수 정답)

11.나 12.나 13.나

FILO 는 First In Last Out  
이므로 LIFO와 같은 방식  
입니다.

## [DB 8강]-자료 구조(선형, 비선형구조)

14. 다음 설명이 의미하는 것은?

- . 삽입과 삭제가 리스트의 양쪽 끝에서 발생할 수 있는 형태이다.
- . 입력이 한쪽에서만 발생하고 출력은 양쪽에서 일어날 수 있는 입력 제한과, 입력은 양쪽에서 일어나고 출력은 한 곳에서만 이루어지는 출력 제한이 있다.

- 가. 스택    나. 큐    다. 이중 스택    라. 데크

15. 선형 자료구조에 해당하지 않는 것은?

- 가. Binary tree    나. Dense list  
다. Doubly linked list    라. Stack

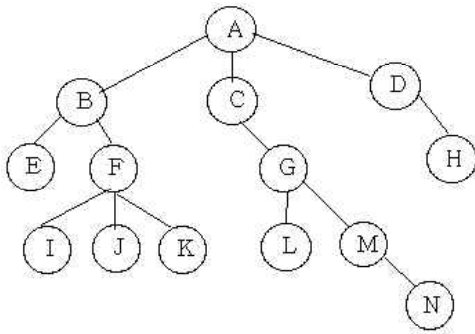
16. 스택(stack)이 사용되는 경우가 아닌 것은?

- 가. 인터럽트의 처리  
나. 수식의 계산  
다. 서브루틴의 복귀번지 저장  
라. 스푼(pool) 처리

[정답] 14.라 15.가 16.라

## [DB 8강]-자료 구조(선형, 비선형구조)

17. 다음과 같은 트리(Tree) 구조에서 기본 용어의 설명이 옳은 것은?



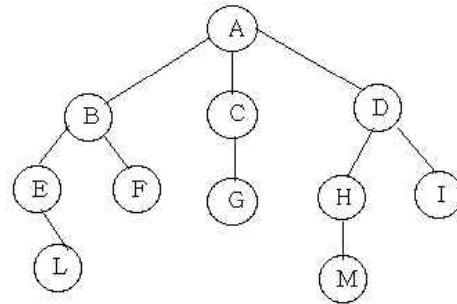
- 가. Node는 10이다.  
 나. Node의 차수(Degree of Node)는 4이다.  
 다. 레벨(Level)은 5이다.  
 라. 근(Root) Node는 N이다.

18. 깊이가 6인 이진 트리의 최대 노드수는?

- 가. 63    나. 64    다. 65    라. 66

[정답] 17.다 18.가 19.나 20.라

19. 다음 Tree의 Degree와 터미널 노드의 수는?



- 가. Degree : 2    터미널 노드 : 4  
 나. Degree : 3    터미널 노드 : 5  
 다. Degree : 4    터미널 노드 : 2  
 라. Degree : 4    터미널 노드 : 10

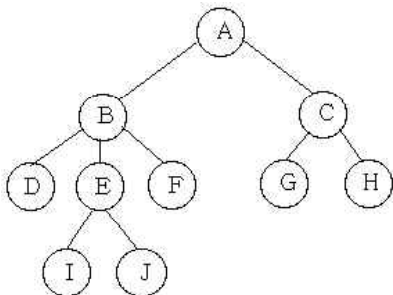
20. 다음과 같은 중위식(Infix)을 후위식(Postfix)으로 올바르게 표현한 것은?

A / B \* (C + D) + E

- 가. + \* / A B + C D E  
 나. C D + A B / \* E +  
 다. A B / (C D +) \* / E +  
 라. A B / C D + \* E +

## [DB 8강]-자료 구조(선형, 비선형구조)

21. 아래 그림에서 트리의 차수(Degree)를 구하면?



- 가. 2    나. 3    다. 4    라. 5

22. 이진 트리의 특성에 대한 설명으로 옳지 않은 것은?

(단,  $n_0$  =단말 노드수,  $n_1$  =차수 1인 노드수,  
 $n_2$  =차수 2인 노드수,  $n$  =노드의 총수,  $e$  =간선의 총수)

- 가.  $n = e + 1$   
 나.  $e = n_1 + 2n_2$   
 다.  $n = n_0 + n_1 + n_2$   
 라.  $n_0 = n_2 + 2$

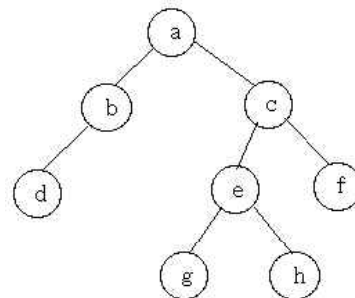
[정답] 21.나 22.라 23.나 24.가

23. 다음과 같이 주어진 후위 표기 방식의 수식을 중위 표기 방식으로 나타낸 것은?

A B C - / D E F + \* +

- 가. A / (B - C) + F \* E + D  
 나. A / (B - C) + D \* (E + F)  
 다. A / (B - C) + D + E \* F  
 라. A / (B - C) \* D + E + F

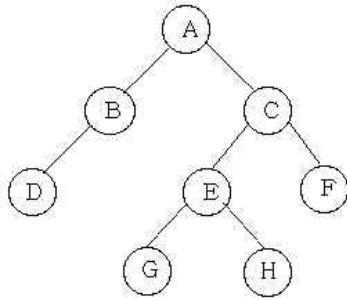
24. 아래 트리 구조에 대하여 Preorder 순서로 처리한 결과는?



- 가. a→b→d→c→e→g→h→f  
 나. d→b→g→h→e→f→c→a  
 다. a→b→c→d→e→f→g→h  
 라. a→b→d→g→e→h→c→f

## [DB 8강]-자료 구조(선형, 비선형구조)

25. 아래 이진 트리를 후위 순서(Postorder)로 운행한 결과는?

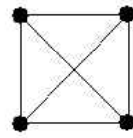


- 가. ABCDEFGH    나. DBGHEFCA  
다. ABDCEGHF    라. BDGHEFAC

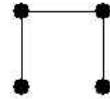
26. 이진 트리(Binary Tree)에서 발생하는 널(Null) 링크를 트리 운행에 필요한 다른 노드의 포인터로 사용하도록 고안된 트리는?

- 가. Knuth 이진 트리(Knuth Binary Tree)  
나. 전 이진 트리(Complete Binary Tree)  
다. B+ 트리(B+ 트리)  
라. 스레드 이진 트리(Threaded Binary Tree)

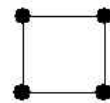
27. 다음 그래프 중 보기의 신장 트리(Spanning Tree)가 아닌 것은?



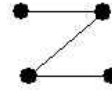
가.



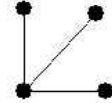
다.



나.



라.



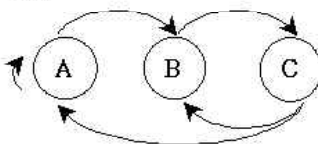
[정답] 25.나 26.라 27.다

## [DB 8강]-자료 구조(선형, 비선형구조)

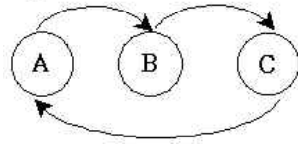
28. 다음 인접행렬(Adjacency Matrix)에 대응되는 그래프(Graph)를 그렸을 때 옳은 것은?

|   | A | B | C |
|---|---|---|---|
| A | 0 | 1 | 0 |
| B | 0 | 0 | 1 |
| C | 1 | 0 | 0 |

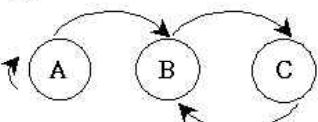
가.



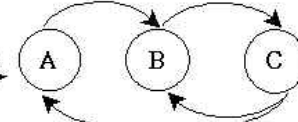
나.



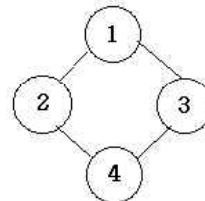
다.



라.



29. 보기와 같은 그래프에서 인접행렬이 옳게 된 것은?



가.

$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

다.

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

나.

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

라.

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

[정답] 28.나 29.라

## [DB 8강]-자료 구조(선형, 비선형구조)

[산-08년3월]

30. 큐(Queue)에 대한 설명으로 옳지 않은 것은?

- 가. 입력은 리스트의 한끝에서, 출력은 그 상대편 끝에서 일어난다.
- 나. 운영체제의 작업 스케줄링에 사용된다.
- 다. 오버플로우는 발생할 수 있어도 언더플로우는 발생되지 않는다.
- 라. 가장 먼저 삽입된 자료가 가장 먼저 삭제되는 FIFO 방식으로 처리된다.

[기-08년3월]

32. 순서가 A, B, C, D로 정해진 입력 자료를 스택에 입력하였다가 출력하는 경우, 출력 결과로서 가능하지 않은 것은?

- 가. D, A, B, C      나. B, D, C, A
- 다. C, B, D, A      라. B, A, D, C

[산-08년3월]

31. 다음의 산술식을 “Postfix” 표기로 옳게 나타낸 것은?

$$X=A+(B+C/D)\times E-F$$

- 가.  $X=A+B+C/D\times E-F$       나.  $XABCD/+E\times+F--$
- 다.  $=X--A\times+B/CDEF$       라.  $XABCDEF=++/\times-$

[정답] 30.다 31.나 32.가

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

[출제빈도 '하']

### 1. 정렬 ★★★★★

: 오름차순 또는 내림차순으로 데이터를 나열함

1) 내부정렬 : 주기억장치 사용

- 선택 정렬 (selection sort), 버블 정렬 (bubble sort), 삽입 정렬 (insertion sort), 힙 정렬 (heap sort)
- 셸 정렬 (shell sort), 퀵 정렬 (quick sort), 2-Way 병합 정렬 (2-way merge sort), 기수 정렬 (radix sort)

2) 외부정렬 : 보조기억장치 사용

- 밸런스드 병합 정렬 (Balanced Merge sort), 캐스캐이드 병합 정렬 (Cascade Merge sort)
- 폴리파즈 병합 정렬 (Polyphase Merge sort), 오실레이팅 병합 정렬 (Oscillation Merge sort)

\* 병합정렬이라는 말을 빼고, 보기를 주는 경우가 있음 (밸런스드 병합정렬 → 밸런스드 정렬)

### 2. 정렬 알고리즘 선택 시 고려사항 ★☆☆☆☆

- 데이터 양
- 초기 데이터의 배열상태
- 키 값들의 분포상태
- 소요공간 및 작업시간
- 운영체제의 종류, 액세스 빈도, 증가 데이터의 배열 상태(X)

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

### 3. 선택정렬(Selection Sort) ★☆☆☆☆

→ 첫 자리부터 정렬하기 (비교, 교환)

점수 100 70 90 80 90      .....▶      70 100 90 80 90 (Pass 1)

.....▶ 70 80 100 90 90 (Pass 2)

.....▶ 70 80 90 100 90 (Pass 3)

.....▶ 70 80 90 90 100 (Pass 4)

|            |     |     |    |    |    |
|------------|-----|-----|----|----|----|
| Pass 1 일 때 | 100 | 70  | 90 | 80 | 90 |
| 교 환        | 70  | 100 | 90 | 80 | 90 |
| 변화 없음      | 70  | 100 | 90 | 80 | 90 |
| 변화 없음      | 70  | 100 | 90 | 80 | 90 |
| 변화 없음      | 70  | 100 | 90 | 80 | 90 |

|            |    |    |     |     |    |
|------------|----|----|-----|-----|----|
| Pass 3 일 때 | 70 | 80 | 100 | 90  | 90 |
| 교 환        | 70 | 80 | 90  | 100 | 90 |
| 변화 없음      | 70 | 80 | 90  | 100 | 90 |

|            |    |     |     |    |    |
|------------|----|-----|-----|----|----|
| Pass 2 일 때 | 70 | 100 | 90  | 80 | 90 |
| 교 환        | 70 | 90  | 100 | 80 | 90 |
| 교 환        | 70 | 80  | 100 | 90 | 90 |
| 변화 없음      | 70 | 80  | 100 | 90 | 90 |

|            |    |    |    |     |     |
|------------|----|----|----|-----|-----|
| Pass 4 일 때 | 70 | 80 | 90 | 100 | 90  |
| 교 환        | 70 | 80 | 90 | 90  | 100 |

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

#### 4. 버블정렬(Bubble Sort) ★☆☆☆☆

→ 수면 위로 올라가는 물방울 (인접 데이터 비교)

점수 100 70 90 80 90      .....➡      70 90 80 90 **100** (Pass 1)

.....▶ 70 80 90 90 100 (Pass 2)

.....▶ 70 80 90 90 100 (Pass 3)

.....▶ 70 80 90 90 100 (Pass 4)

|            |  |     |     |     |     |     |
|------------|--|-----|-----|-----|-----|-----|
| Pass 1 일 때 |  | 100 | 70  | 90  | 80  | 90  |
| 교 환        |  | 70  | 100 | 90  | 80  | 90  |
| 교 환        |  | 70  | 90  | 100 | 80  | 90  |
| 교 환        |  | 70  | 90  | 80  | 100 | 90  |
| 교 환        |  | 70  | 90  | 80  | 90  | 100 |

|            |    |    |    |    |     |
|------------|----|----|----|----|-----|
| Pass 3 일 때 | 70 | 80 | 90 | 90 | 100 |
| 변화 없음      | 70 | 80 | 90 | 90 | 100 |
| 변화 없음      | 70 | 80 | 90 | 90 | 100 |

|            |    |    |    |    |     |
|------------|----|----|----|----|-----|
| Pass 2 일 때 | 70 | 90 | 80 | 90 | 100 |
| 변화 없음      | 70 | 90 | 80 | 90 | 100 |
| 교 환        | 70 | 80 | 90 | 90 | 100 |
| 변화 없음      | 70 | 80 | 90 | 90 | 100 |

|            |    |    |    |    |     |
|------------|----|----|----|----|-----|
| Pass 4 일 때 | 70 | 80 | 90 | 90 | 100 |
| 변화 없음      | 70 | 80 | 90 | 90 | 100 |

## [DB 9강]-자료 구조(정렬, 검색, 해싱)



### 5. 삽입 정렬(insertion Sort)

→ 성적순으로 교실 자리 배치 하기

점수 100 70 90 80 90      .....→ 70 100 90 80 90 (Pass 1)  
 .....→ 70 90 100 80 90 (Pass 2)  
 .....→ 70 80 90 100 90 (Pass 3)  
 .....→ 70 80 90 90 100 (Pass 4)

#### [70 삽입하기]

Pass 1 일 때      100 70 90 80 90  
 이    동              100 90 80 90  
 삽    입              70 100 90 80 90

#### [90 삽입하기]

Pass 2 일 때      70 100 90 80 90  
 이    동              70      100 80 90  
 변화없음          70      100 80 90  
 삽    입              70 90 100 80 90

#### [80 삽입하기]

Pass 3 일 때      70 90 100 80 90  
 이    동              70 90      100 90  
 이    동              70      90 100 90  
 변화없음          70      90 100 90  
 삽    입              70 80 90 100 90

#### [90 삽입하기]

Pass 4 일 때      70 80 90 100 90  
 이    동              70 80 90      100  
 변화없음          70 80 90      100  
 변화없음          70 80 90      100  
 변화없음          70 80 90      100  
 삽    입              70 80 90 90 100

\* Pass 1에서 '두 번째 값'을 Key 로 선택한다.

## [DB 9강]-자료 구조(정렬, 검색, 해싱)



### 6. 2-Way 병합 정렬

→ 두 개씩 묶어서 정렬

71, 2, 38, 5, 7, 61, 11, 26, 53, 42

1회전 : (71, 2), (38, 5), (7, 61), (11, 26), (53, 42)  
 (2, 71), (5, 38), (7, 61), (11, 26), (42, 53)

2회전 : ((2, 71), (5, 38)), ((7, 61), (11, 26)), (42, 53)  
 (2, 5, 38, 71), (7, 11, 26, 61), (42, 53)

3회전 : ((2, 5, 38, 71), (7, 11, 26, 61)), (42, 53)  
 (2, 5, 7, 11, 26, 38, 61, 71), (42, 53)

4회전 : ((2, 5, 7, 11, 26, 38, 61, 71), (42, 53))  
 2, 5, 7, 11, 26, 38, 42, 53, 61, 71



## [DB 9강]-자료 구조(정렬, 검색, 해싱)

정렬해야 할 파일이(5, 1, 4, 3, 8, 2)인 6개의 키 값을 첫번째 단계에서 3회 수행한 결과가 다음과 같을 때, 어떤 정렬 기법을 사용하였는가? <답은 나>

|                        |                       |
|------------------------|-----------------------|
| 초기 : 5, 1, 4, 3, 8, 2  | 1회 : 1, 5, 4, 3, 8, 2 |
| 2 회 : 1, 4, 5, 3, 8, 2 | 3회 : 1, 4, 3, 5, 8, 2 |

가. 삽입 정렬      나. 버블 정렬      다. 합병 정렬      라. 힙 정렬

→ 초기에서 1회전 후에 상태는 1,4,3,5,2,8 입니다. (문제에서는 단계라고 했으므로 단계=회전 으로 생각하세요.)

1회전을 좀 더 상세히 살펴 보겠습니다. (문제에서는 회라고 했습니다.)

초기 : 5, 1, 4, 3, 8, 2 → 1회 : 1, 5, 4, 3, 8, 2

초기 5,1 을 비교해서 큰 값을 우측으로 이동합니다. 결과 : 1,5

1회 : 1, 5, 4, 3, 8, 2 → 2 회 : 1, 4, 5, 3, 8, 2

1회 5,4 를 비교해서 큰 값을 우측으로 이동합니다. 결과 : 4,5

2 회 : 1, 4, 5, 3, 8, 2 → 3회 : 1, 4, 3, 5, 8, 2

2회 5,3 을 비교해서 큰 값을 우측으로 이동합니다. 결과 : 3,5

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

### 1. 검색

: 원하는 데이터를 찾을 때.

#### 1) 선형 검색 (순차검색)

- 모든 레코드들을 대상으로 순차적 검색, 자료가 정렬되지 않을 때, 처리 속도가 느리다

#### 2) 이분(이진) 검색 ★★☆☆☆☆

- 자료가 정렬되어 있어야 함, 중간 값을 비교 검색, 많은 레코드 검색 시 효율적

|    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|
| 10 | 15 | 31 | 45 | 62 | 63 | 74 | 87 | 91 | 92 |
|----|----|----|----|----|----|----|----|----|----|

#### 4) 보간 검색 : 찾고자 하는 레코드 키가 있음직한 위치를 추정

#### 5) 트리 검색, 블럭 검색, 피보나치 검색

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

### 2. 해싱(hashing) 검색 ★★★★★☆

: 해싱 함수를 이용하여 자료를 검색하는 방법. 데이터를 해시 테이블이라는 배열에 저장하고, 해싱 함수를 이용하여 데이터가 위치한 곳의 주소를 찾기 때문에 신속하게 원하는 자료를 검색할 수 있는 키-주소 변환 방법

#### 1) 특징

- DAM(직접접근, Direct Access Method) 파일을 구성할 때 해싱이 사용
- 삽입, 삭제 작업의 빈도가 많을 때 유리한 방식
- 검색은 가장 빠르지만 기억공간의 낭비 발생

#### 2) 용어 정리

- 해싱 함수 : 해시 테이블의 주소를 생성해 내는 함수
- 해시 테이블 : 해싱 함수에 의하여 참조되는 테이블
- 버킷 (bucket) : 하나의 주소를 갖는 파일의 한 구역
- 슬롯 (slot) : n개의 슬롯이 모여 하나의 버킷을 형성
- 충돌 (collision) : 서로 다른 2개 이상의 레코드가 같은 주소를 갖는 현상
- 시노임 (Synonym): 같은 주소를 갖는 레코드의 집합
- 오버플로 : 버킷 내에 기억 공간이 없는 현상

8 7 6 9 4 14 2 15 19

7로 나눈 후  
나머지 계산

해싱 함수

| 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |

해시 테이블

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

1. 정렬(Sort) 알고리즘의 선택 시 고려 사항으로 거리가 먼 것은?

- 가. 증가 데이터의 배열 상태
- 나. 키 값들의 분포 상태
- 다. 소요공간 및 작업시간
- 라. 정렬에 필요한 기억공간의 크기

2. 주기억장치 내에서 정렬하는 방법에 해당하는 것은?

- 가. Cascade Merge Sort      나. Heap Sort
- 다. Balanced Merge Sort    라. Polyphase Merge Sort

3. 외부 정렬(External Sort)에 해당하지 않는 것은?

- 가. Balanced Sort      나. Cascade Sort
- 다. Insertion Sort    라. Polyphase Sort

4. 인접한 데이터를 비교하면서 그 크기에 따라 데이터의 위치를 바꾸어 정렬하는 방법은?

- 가. 퀵 정렬(Quick Sort)    나. 힙 정렬(Heap Sort)
- 다. 셸 정렬(Shell Sort)    라. 버블 정렬(Bubble Sort)

5. 자료가 아래와 같을 때, 삽입(Insertion) 정렬 방법을 적용하여 오름차순으로 정렬할 경우 Pass 1을 수행한 결과는?

20, 19, 14, 16, 18

- 가. 19, 20, 14, 16, 18      나. 14, 20, 19, 16, 18
- 다. 14, 19, 20, 16, 18    라. 20, 14, 19, 16, 18

6. 자료가 아래와 같이 주어졌을 때, 선택 정렬(Selection Sort)을 적용하여 오름차순으로 정렬할 경우 Pass 2를 진행한 후의 정렬된 값으로 옳은 것은?

9, 4, 5, 11, 8

- 가. 4, 5, 9, 8, 11
- 나. 4, 5, 9, 11, 8
- 다. 4, 5, 8, 11, 9
- 라. 4, 5, 8, 9, 11

[정답] 1.가 2.나 3.다 4.라 5.가 6.나

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

7. 레코드의 많은 자료 이동을 없애고 하나의 파일을 부분적으로 나누어가면서 정렬하는 방법으로 키를 기준으로 작은 값은 왼쪽에 큰 값은 오른쪽에 모이도록 서로 교환시키는 부분 교환 정렬법은?

- 가. 퀵 정렬(Quick Sorting)
- 나. 셸 정렬(Shell Sorting)
- 다. 삽입 정렬(Insertion Sorting)
- 라. 선택 정렬(Selection Sorting)

8. 이진 트리의 레코드 R=(88, 74, 63, 55, 37, 25, 33, 19, 26, 14, 9)에 대하여 힙(Heap) 정렬을 만들 때, 37의 왼쪽과 오른쪽의 자노드(Child Node)의 값은?

- 가. 55, 25                      나. 63, 33
- 다. 33, 19                      라. 14, 9

9. 연산 시간이 최악과 평균의 경우, 모두  $O(n \log n)$ 으로 빠른 속도를 갖는 정렬 방식은?

- 가. 퀵 정렬(quick sort)      나. 버블 정렬(bubble sort)
- 다. 힙 정렬(heap sort)      라. 선택 정렬(selection sort)

10. 다음은 어떠한 정렬 방법을 설명한 것인가?

- . 두 개의 키들을 한 쌍으로 하여 각 쌍에 대하여 순서를 정한다.
- . 순서대로 정렬된 각 쌍의 키들을 합병하여 하나의 정렬된 서브리스트로 만든다.
- . 위 과정에서 정렬된 서브리스트들을 하나의 정렬된 파일이 될 때까지 반복한다.

- 가. 2-Way 합병 정렬
- 나. 퀵 정렬
- 다. 기수 정렬
- 라. 버블 정렬

11. 정렬 알고리즘 선택 시 고려하여야 할 사항으로 거리가 먼 것은?

- 가. 데이터의 양
- 나. 초기 데이터의 배열상태
- 다. 키 값들의 분포상태
- 라. 운영체제의 종류

[정답] 7.가 8.라 9.다 10.가 11.라

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

12. 다음과 같이 레코드가 구성되어 있을 때, 이진 검색 방법으로 14를 찾을 경우 비교되는 횟수는?

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

- 가. 2번    나. 3번    다. 4번    라. 5번

13. 아래 보기의 자료에서 이진 탐색(Binary Search)을 적용할 경우 m을 찾기 위한 비교 횟수는?

a b c d e f g h i j k l m n o

- 가. 3    나. 4    다. 5    라. 6

14. 해시(Hash) 함수와 밀접한 관계가 있는 파일은?

- 가. ISAM 파일              나. VSAM 파일
- 다. DAM 파일              라. 링 파일

15. 해싱 함수의 값을 구한 결과 키 k1, k2가 같은 값을 가질 때, 이들 키 k1, k2의 집합을 무엇이라 하는가?

- 가. Mapping    나. Folding    다. Synonym    라. Chaining

16. 키 값을 여러 부분으로 분류하여 분류한 부분을 더하거나 XOR하여 주소를 계산하는 해싱 함수의 종류는?

- 가. 제산(Divide)함수
- 나. 접지(Folding)함수
- 다. 중간제곱(Mid-Square)함수
- 라. 숫자 분석 함수

17. 버킷(bucket)과 가장 관련이 깊은 것은?

- 가. SAM    나. ISAM    다. B-Tree    라. Hashing

18. 해싱함수(Hashing Function)의 종류가 아닌 것은?

- 가. 제곱(Mid-Square) 방법
- 나. 숫자분석(Digit Analysis) 방법
- 다. 체인(Chain) 방법
- 라. 제산(Division) 방법

19. 검색 방법 중 속도는 가장 빠르지만 충돌현상 시 오버플로 해결의 부담이 과중되며, 많은 기억공간을 요구하는 탐색 방법은?

- 가. 해싱(hashing)
- 나. 블록 탐색(block search)
- 다. 순차 탐색(sequential search)
- 라. 이진 탐색(binary search)

[정답] 12.나 13.나 14.다 15.다 16.나 17.라 18.다 19.가

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

20. 해싱(hashing)에서 서로 다른 키(key)가 같은 홈주소(home address)를 가지는 경우를 무엇이라 하는가?

- 가. 동의어(synonym)
- 나. 재귀(recursion)
- 다. 충돌(collision)
- 라. 버킷(bucket)

22. 다음의 자료를 삽입(insert) 정렬 기법을 사용하여 오름차순으로 정렬할 경우 PASS 2 의 결과는?

64 28 33 76 55 12 43

- 가. 28 64 33 76 55 12 43
- 나. 12 28 33 55 64 76 43
- 다. 28 33 64 76 55 12 43
- 라. 12 64 28 33 76 55 43

21. 해싱(Hashing)에 관한 설명으로 옳지 않은 것은?

- 가. 버킷(bucket)이란 하나의 주소를 갖는 파일의 한 구역을 의미하며, 버킷의 크기는 같은 주소에 포함될 수 있는 레코드 수를 의미한다.
- 나. 슬롯(slot)이란 한 개의 레코드를 저장할 수 있는 공간으로 n개의 슬롯이 모여 하나의 버킷을 형성한다.
- 다. 충돌(collision)이란 레코드를 삽입할 때 2개의 상이한 레코드가 똑같은 버킷으로 해싱되는 것을 의미한다.
- 라. 해싱은 충돌(collision)이 발생하면 항상 오버플로우(overflow)가 발생한다.

[정답] 20.다 21.라 22.다

## [DB 9강]-자료 구조(정렬, 검색, 해싱)

[산-08년3월]

23. 해싱 함수 기법에서 키 값을 양의 정수인 소수로 나누어 나머지를 홈 주소로 취하는 방법을 무엇이라고 하는가?

- 가. 폴딩(Folding)법      나. 제곱(Mid-Square)법
- 다. 제산(Division)법      라. 기수(Radix)변환법

[기-08년5월][산-08년3월]

24. 해싱 기법에서 동일한 홈 주소로 인하여 충돌이 일어난 레코드들의 집합을 무엇이라고 하는가?

- 가. Synonym      나. Collision
- 다. Bucket      라. Overflow

[정답] 23.다 24.가

[해싱함수 종류]

- 기수(Radix)법 : 다른 진법으로 변환
- **folding(중첩)법** : 더하거나 XOR
- 계수 분석법 : 숫자의 분포 조사
- 제산(Division)법 : 나누어 나머지
- 제곱법 : 제곱

[오버플로우 해결 방법]

- 개방주소법 : 다음 빈 버킷에 저장
- 폐쇄주소법 : 별도의 공간에 저장
- 재해싱법 : 새로운 해싱함수를 적용



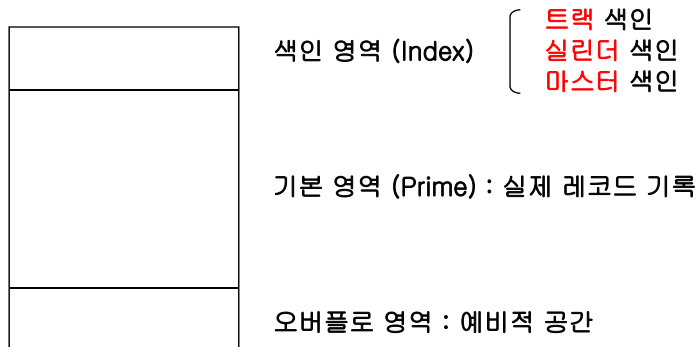
## 1. 순차 파일(SAM : Sequential Access Method) : 목차 없는 책

- : 파일 내의 각 레코드를 논리적 순서에 따라 물리적으로 연속된 위치에 기록한 파일
- 기억장소의 낭비가 없다.
- 색인순차파일에 비해 삽입, 삭제, 검색이 어렵다.



## 2. 색인 순차 파일(ISAM : Index) : 목차 있는 책 (정적 인덱스)

- : 인덱스를 통한 랜덤 처리와 데이터의 순차 처리를 병행할 수 있는 파일
- 삽입, 삭제, 갱신, 검색 용이
- 삽입시 기본 영역에 추가 공간이 없을 경우 오버플로 영역에 저장
- 재사용이 안되므로 삽입, 삭제가 빈번할 경우 기억공간 낭비 발생하므로 재구성이 필요

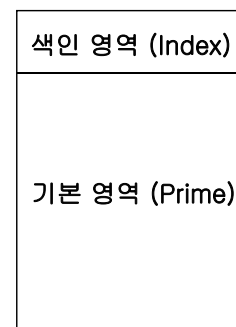


## 3. 직접 파일(DAM : Direct) ★☆☆☆☆

- : 해싱 함수를 계산해서 물리적 주소를 직접 접근 (대화형 처리 가능)
- 순서에 관계없이 저장
- 레코드 주소의 변환과정의 시간 소요
- 기억공간 효율 저하

## 4. VSAM(Virtual : 동적 인덱스) ★☆☆☆☆

- : 동적 인덱스 방법을 이용한 색인 순차 파일
- 기본 구역과 오버플로우 구역을 구분하지 않음  
(기본 구역내에 예비 공간을 두어 추가로 삽입될 경우 이용)
- 레코드를 삭제하면 그 공간을 재사용할 수 있음  
(정적 인덱스는 재사용 안됨)
- 제어 구간에 가변 길이 레코드를 쉽게 수용할 수 있음



## 5. 역 파일 : 찾아보기 기능 ★☆☆☆☆

- : 특정파일을 여러 개의 색인으로 만들어 항목별 특성에 맞게 작업하도록 구성
- 질의응답 시간 단축되고 처리가 쉽다.
- 색인의 각 항목의 길이가 가변

## [DB 10강]-자료 구조(파일편성, 인덱스)

### 6. 인덱스 ★☆☆☆☆

- 인덱스를 통해서 테이블의 레코드에 대한 액세스를 빠르게 수행할 수 있다.
- 인덱스는 하나 이상의 필드로 만들어도 된다
- 레코드의 삽입과 삭제가 수시로 일어나는 경우는 인덱스를 최소화한다

## [DB 10강]-자료 구조(파일편성, 인덱스)

1. 색인 순차 파일(Indexed Sequential Access Method File)의 인덱스에 해당하지 않는 것은?

- 가. Master 인덱스    나. Prime 인덱스  
다. Cylinder 인덱스    라. Track 인덱스

2. 인덱스나 데이터 파일을 블록으로 구성하고 각 블록에는 추가로 삽입될 레코드를 감안하여 빈 공간을 미리 예비해 두는 인덱스 방법은?

- 가. 정적 인덱스 방법    나. 동적 인덱스 방법  
다. 집중화 인덱스 방법    라. 보조 인덱스 방법

3. 색인 순차 파일(ISAM, Indexed Sequential Access Method)에 관한 설명으로 옳지 않은 것은?

- 가. 순차 처리와 랜덤 처리가 모두 가능하다.  
나. 레코드를 추가 및 삽입하는 경우, 파일 전체를 복사할 필요가 없다.  
다. 기본 구역(Prime Data Area), 색인 구역(Index Area), 오버플로 구역(Overflow Area)으로 구성되어 있다.  
라. 해시 함수를 사용하여 레코드를 저장할 위치를 결정한다.

4. VSAM 파일에 대한 설명으로 거리가 먼 것은?

- 가. 기본 데이터 영역과 오버플로 영역을 구분하지 않는다.  
나. 레코드를 삭제하면 그 공간을 재사용할 수 있다.  
다. 제어 구간에 가변 길이 레코드를 쉽게 수용할 수 있다.  
라. 특정 레코드에 대해 빠르고 직접적인 접근을 지원할 수 있기 때문에 대화형 처리에 많이 이용된다.

5. 파일에 대한 설명 중 옳지 않은 것은?

- 가. 순차 파일(Sequential File)은 생성되는 순서에 따라 레코드를 순차적으로 저장하므로, 저장 매체의 효율이 가장 높다.  
나. 직접 파일(Direct File)은 특정 레코드에 접근하기 위해서 디스크의 물리적인 주소로 변환할 수 있는 함수를 사용한다.  
다. 색인 순차 파일(Indexed Sequential File)은 순차 및 직접 접근 형태를 모두 지원할 수 있으나, 기억 장소의 낭비를 초래한다.  
라. VSAM 파일(Virtual Storage Access Method File)은 검색 속도를 빠르게 하기 위하여, 기본 데이터 구역과 오버플로 구역을 구분하여 갖추어야 한다.

[정답] 1.나 2.나 3.라 4.라 5.라