

제3과목 데이터베이스 구축

01 논리 데이터베이스 설계 A



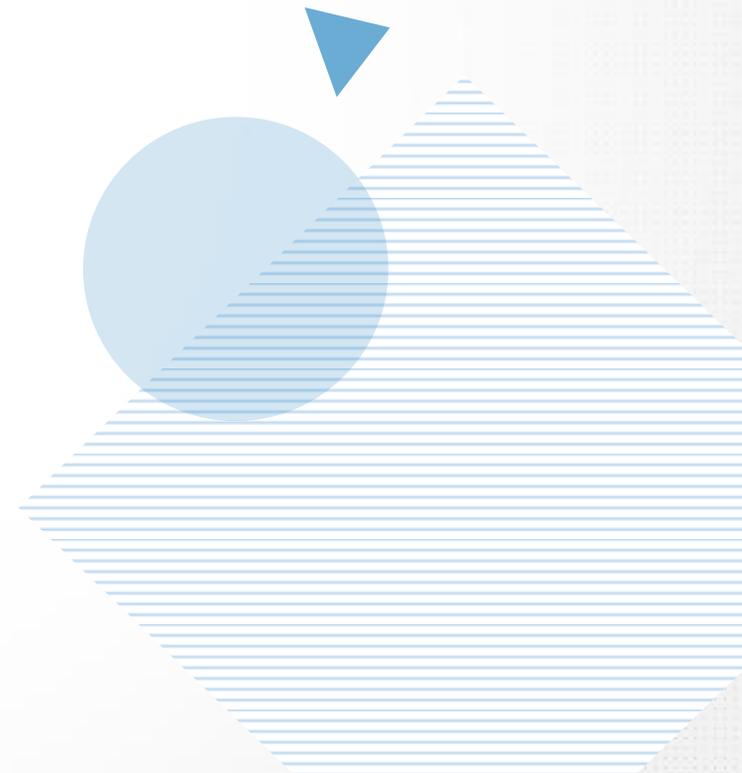
데이터베이스 설계

- 데이터베이스 설계의 개념
- 데이터베이스 설계 시 고려사항
 - 무결성, 일관성, 회복, 보안, 효율성, 데이터베이스 확장
- 데이터베이스 설계 순서
 - 요구조건 분석 - 개념적 설계 - 논리적 설계 - 물리적 설계 - 구현



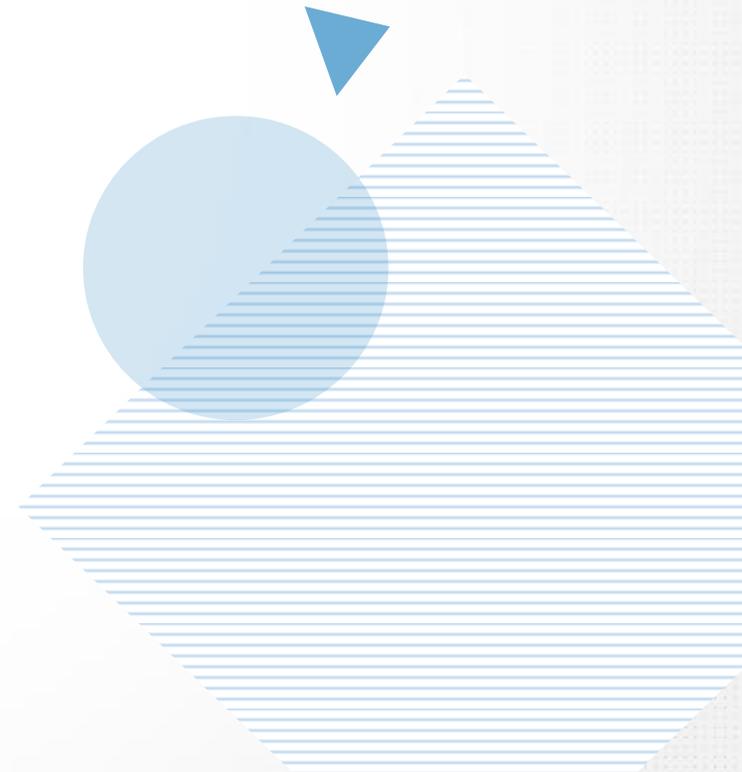
🔗 데이터베이스 설계

- 요구 조건 분석
- 개념적 설계(정보 모델링, 개념화)
- 논리적 설계(데이터 모델링)
- 물리적 설계(데이터 구조화)
- 데이터베이스 구현



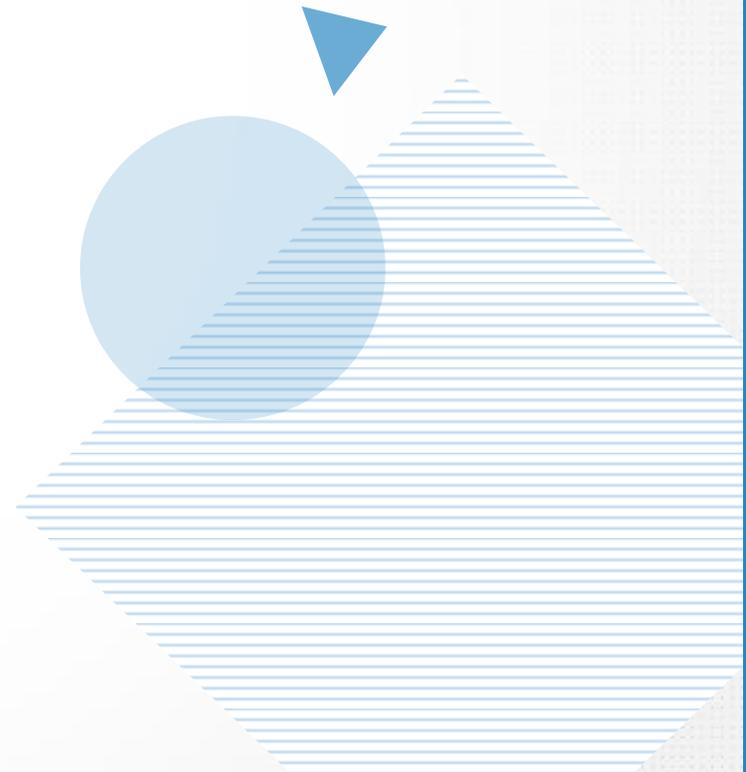
데이터 모델의 개념

- 데이터 모델의 정의
- 데이터 모델의 구성 요소
 - 개체, 속성, 관계
- 개념적 데이터 모델
- 논리적 데이터 모델
- 논리적 데이터 모델의 품질 검증
 - 개체, 속성, 관계, 식별자, 전반적인 품질 검증항목
- 데이터 모델에 표시할 요소
 - 구조, 연산, 제약조건



🔗 데이터 모델의 구성 요소 - 개체(Entity)

- ▶ 개체의 정의 및 특징
- ▶ 개체 선정 방법
- ▶ 개체명 지정 방법

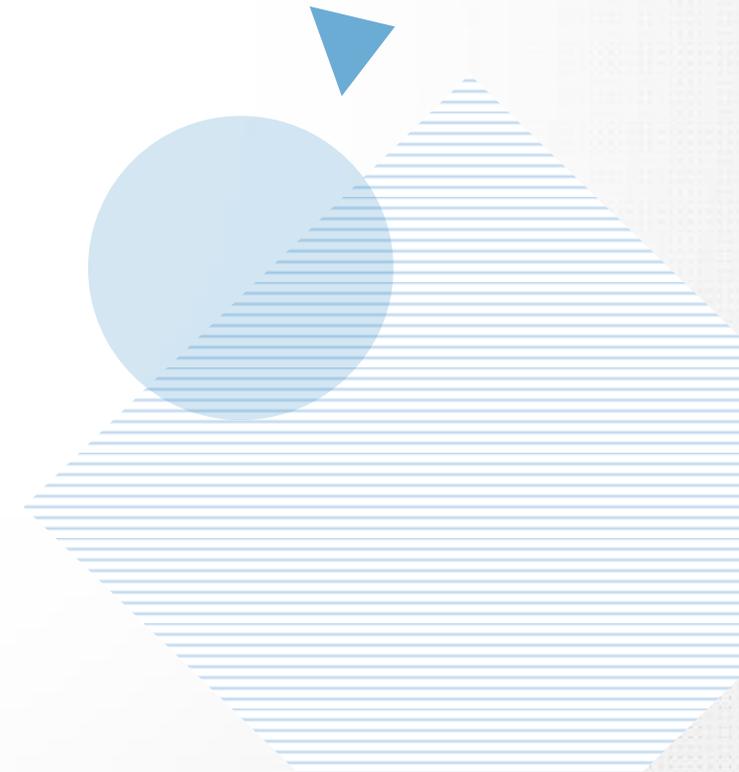


1. 다음 괄호안에 들어갈 내용으로 타당한 것은?

데이터 모델은 일반적으로 3가지 구성요소를 포함하고 있다.
데이터구조, (), 제약조건이다.

- ① 개체
- ② 연산
- ③ 속성
- ④ 도메인

정답 2



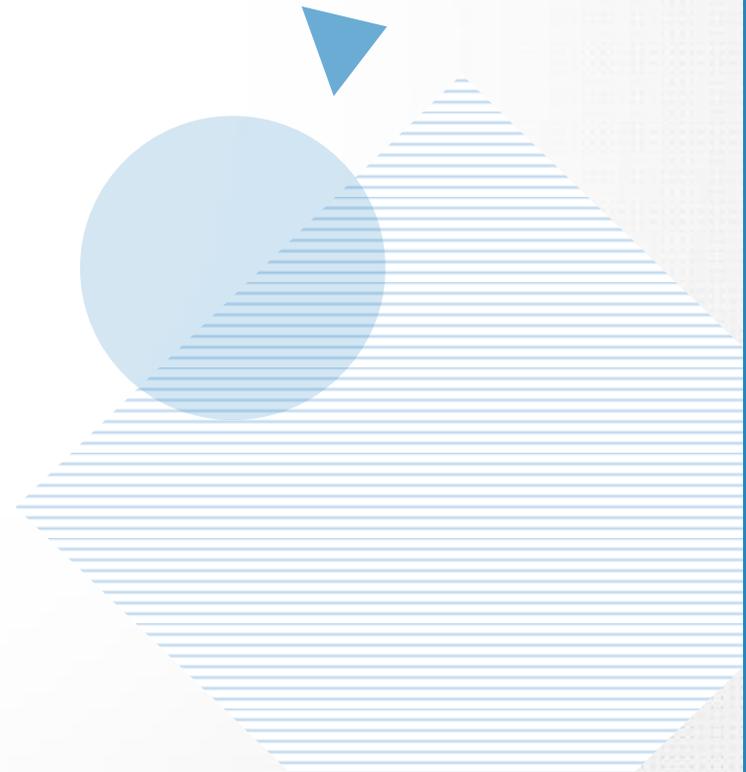
제3과목 데이터베이스 구축

02 논리 데이터베이스 설계 B



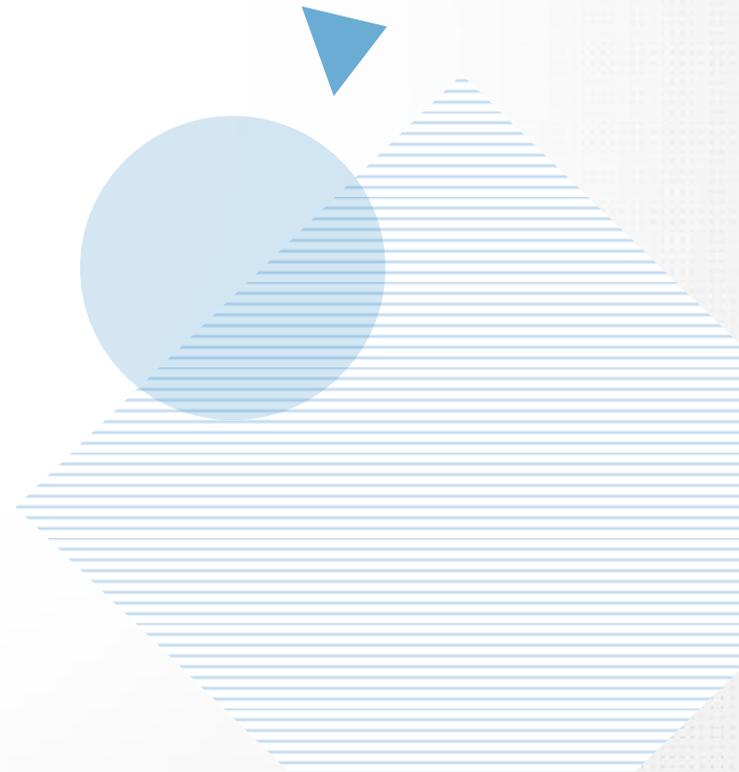
데이터 모델의 구성 요소 - 속성(Attribute)

- ▶ 속성의 정의 및 특징
- ▶ 속성의 종류
 - 속성의 특성에 따른 분류 : 기본속성, 설계속성, 파생속성
 - 개체 구성방식에 따른 분류 : 기본키 속성, 외래키 속성, 일반 속성
- ▶ 속성 후보 선정 원칙
- ▶ 속성명 지정 원칙



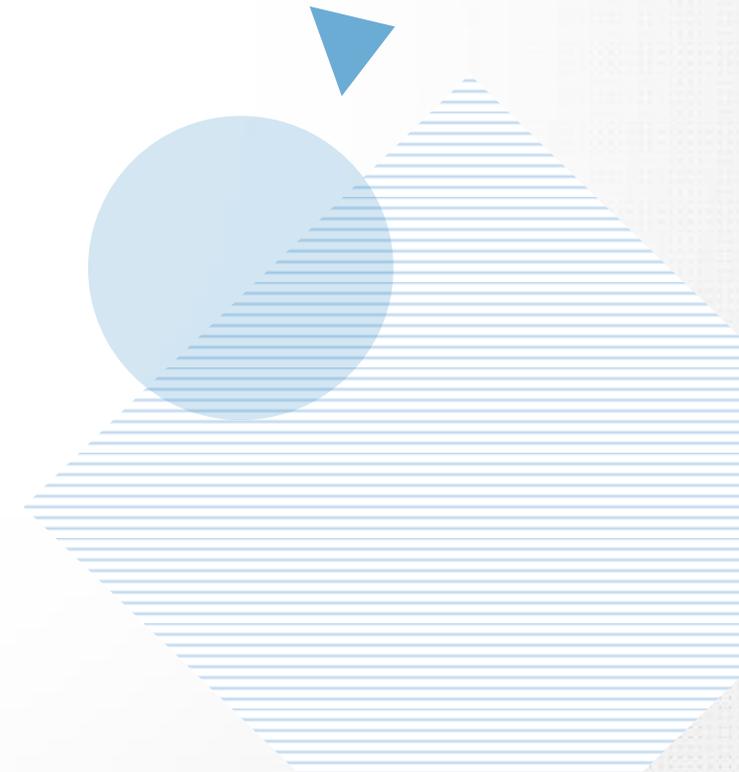
데이터 모델의 구성 요소 - 관계(Relationship)

- ▶ 관계의 정의
- ▶ 관계의 형태
 - 일대일, 일대다, 다대다
- ▶ 관계의 종류
 - 종속 관계, 중복 관계, 재귀 관계, 배타 관계



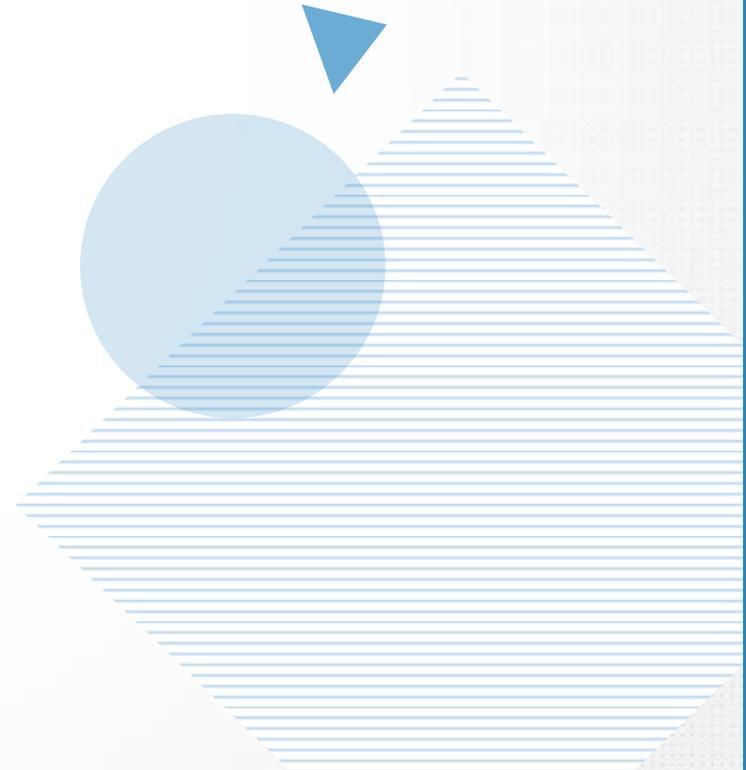
▶ 식별자의 정의

- 대표성 여부 : 주 식별자, 보조 식별자
- 스스로 생성 여부 : 내부 식별자, 외부 식별자
- 단일 속성 여부 : 단일 식별자, 복합 식별자
- 대체여부 : 원조 식별자, 대리 식별자



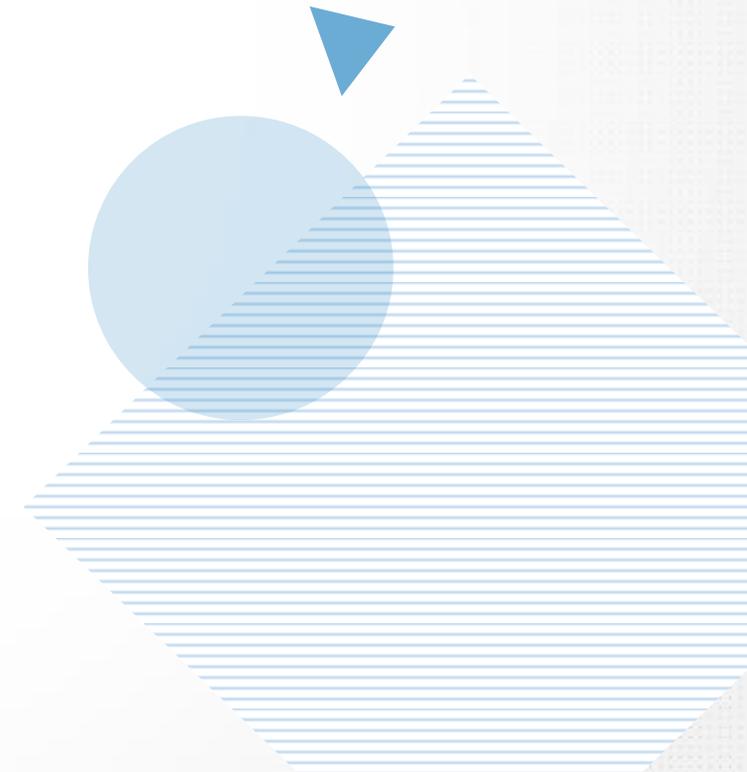
식별자

- 주 식별자/보조 식별자
 - 주 식별자의 4가지 특성 : 유일성, 최소성, 불변성, 존재성
- 내부 식별자/외부 식별자
- 단일 식별자/복합 식별자
- 원조 식별자/대리 식별자
- 후보 식별자



E-R(개체-관계) 모델

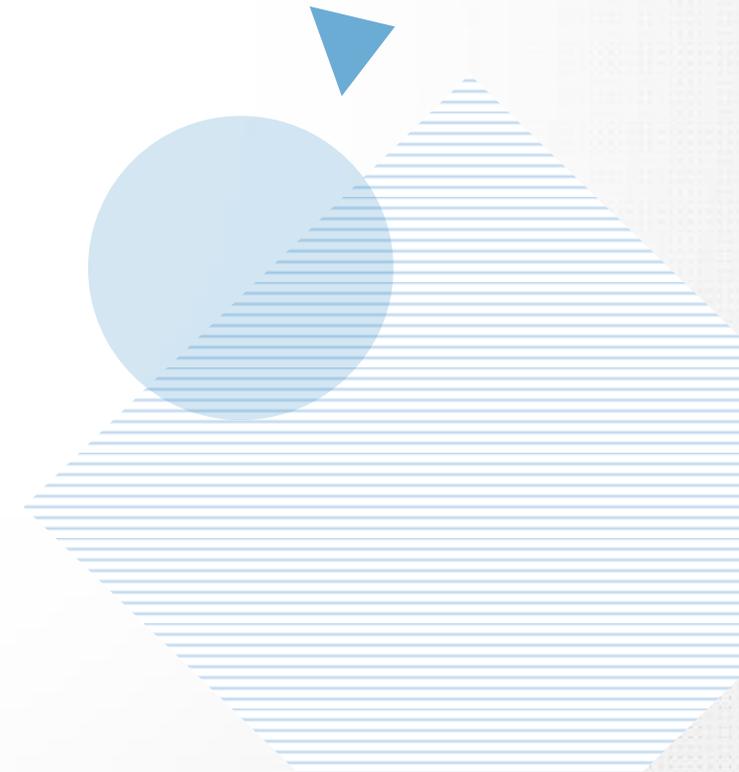
- E-R(Entity-Relationship, 개체-관계) 모델의 개요
- E-R 다이어그램
 - 피터 첸 표기법
 - 정보 공학 표기법(Information Engineering Notation)
 - 바커 표기법(Barker Notation)



1. E-R 다이어그램의 구성요소에 대한 표현이 틀리게 짝지어진것은 ?

- ① 개체집합-직사각형
- ② 관계집합-마름모꼴
- ③ 속성-원
- ④ 링크-화살표

정답4



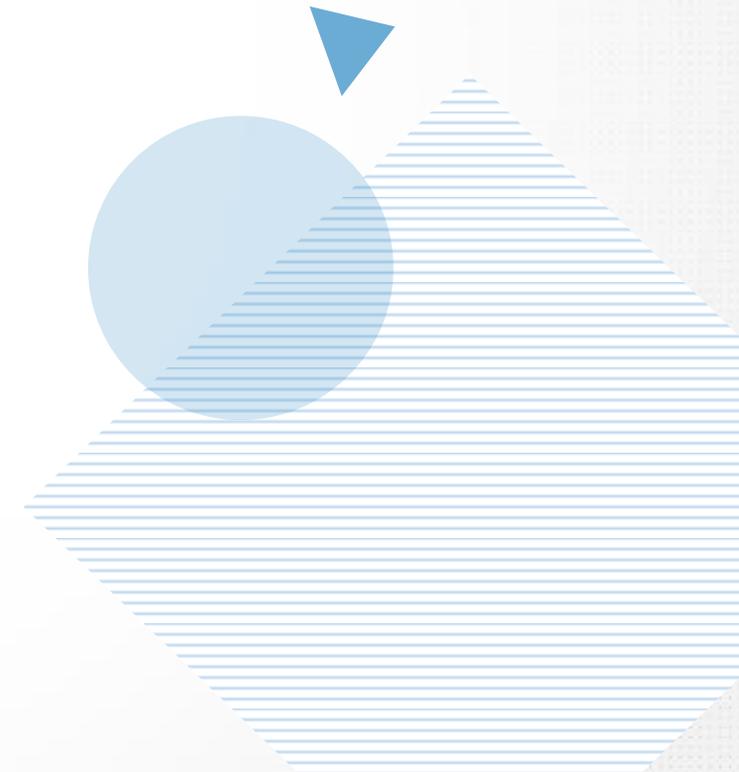
제3과목 데이터베이스 구축

03 논리 데이터베이스 설계 C



🔗 관계형 데이터 모델

- 관계형 데이터 모델(Relational Data Model)의 개요
- 관계형 데이터 모델의 구성



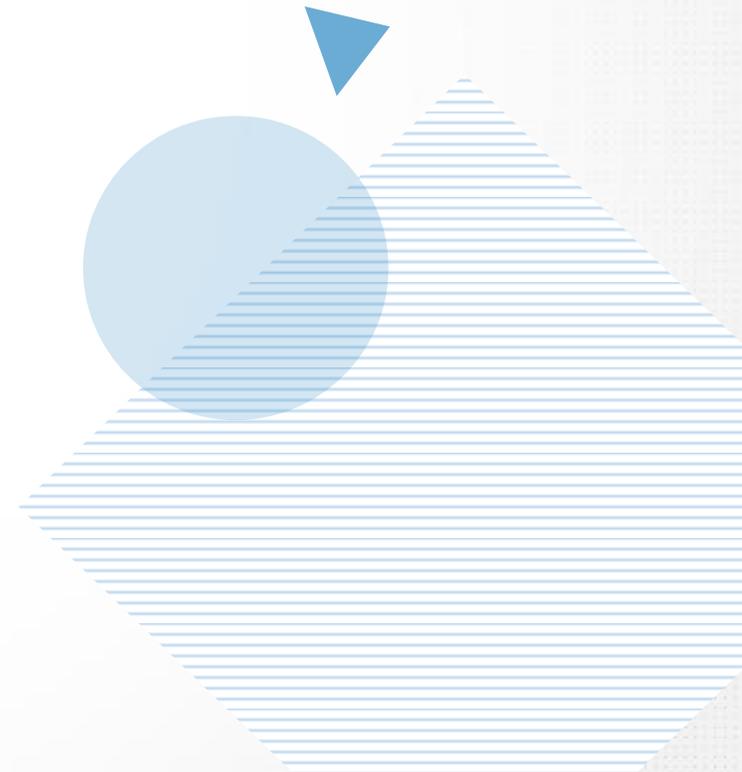
🔄 관계형 데이터베이스의 구조

- 관계형 데이터베이스의 개요
- 관계형 데이터베이스의 Relation 구조
 - 튜플
 - 속성
 - 도메인
- 릴레이션의 특징



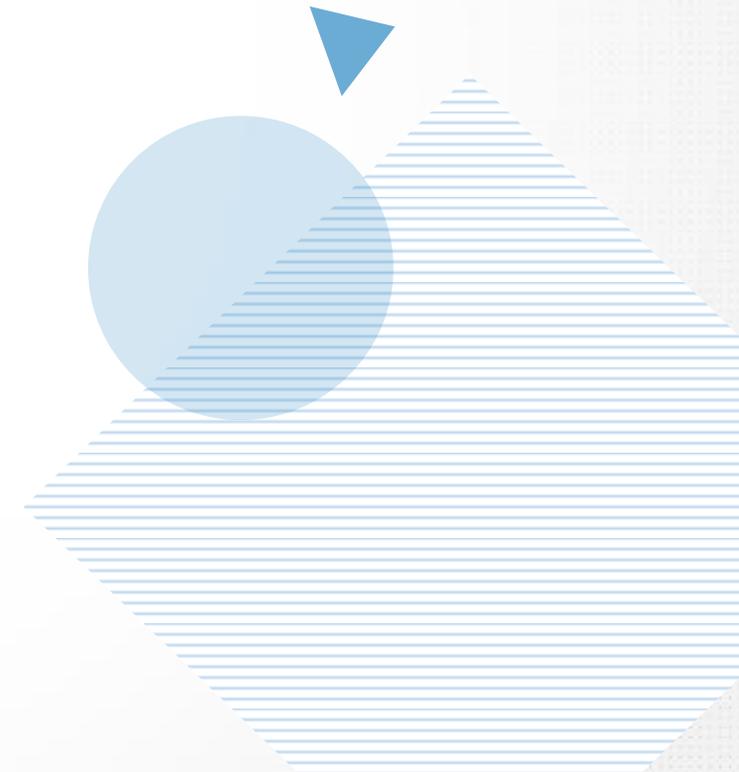
🔗 관계형 데이터베이스의 제약 조건 - 키(Key)

- 키(Key)의 개념
- 후보키(Candidate Key)
- 기본키(Primary Key)
- 대체키(Alternate Key)
- 슈퍼키(Super Key)
- 외래키(Foreign Key)



🔗 관계형 데이터베이스의 제약 조건 - 무결성

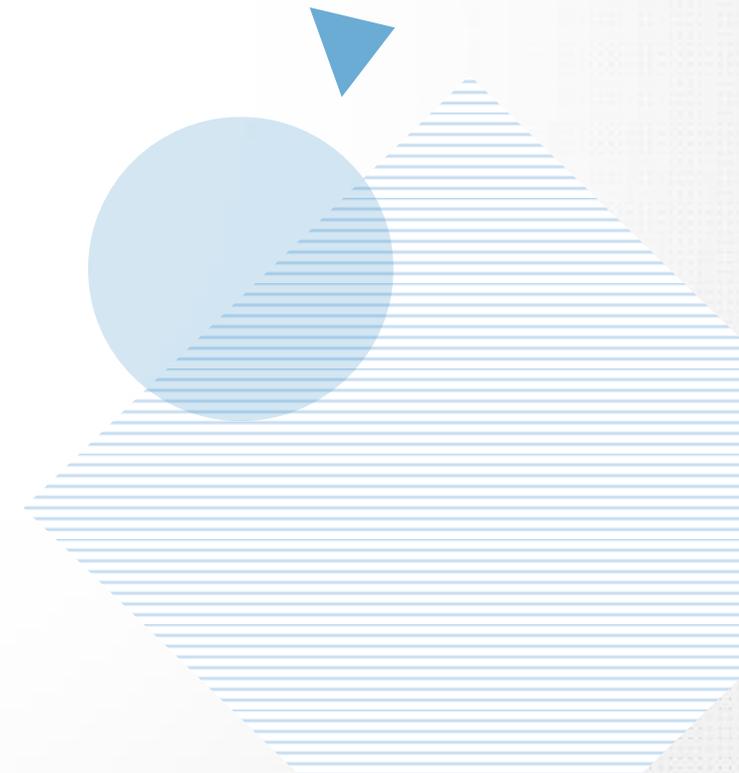
- 무결성(Integrity)의 개념
- 개체 무결성(Entity Integrity, 실체 무결성)
- 도메인 무결성(Domain Integrity, 영역 무결성)
- 참조 무결성(Referential Integrity)
- 사용자 정의 무결성
- 데이터 무결성 강화
 - 애플리케이션, 데이터베이스 트리거, 제약조건



1. 한 릴레이션의 기본키를 구성하는 어떠한 속성값도 널값이나 중복값을 가질 수 없다는 것을 의미하는 것은?

- ① 개체 무결성 제약 조건
- ② 참조 무결성 제약 조건
- ③ 보안 무결성 제약 조건
- ④ 정보 무결성 제약 조건

정답 1

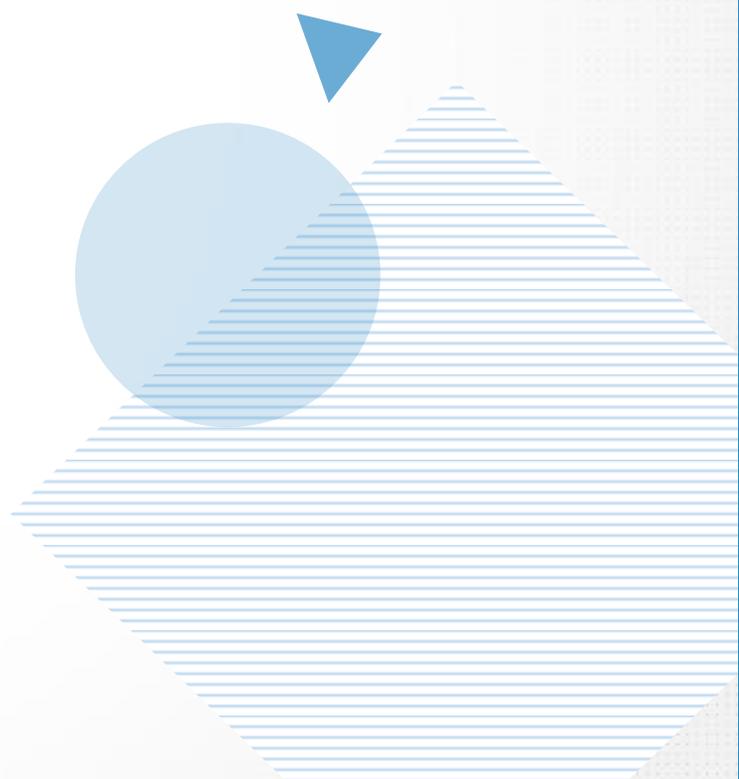
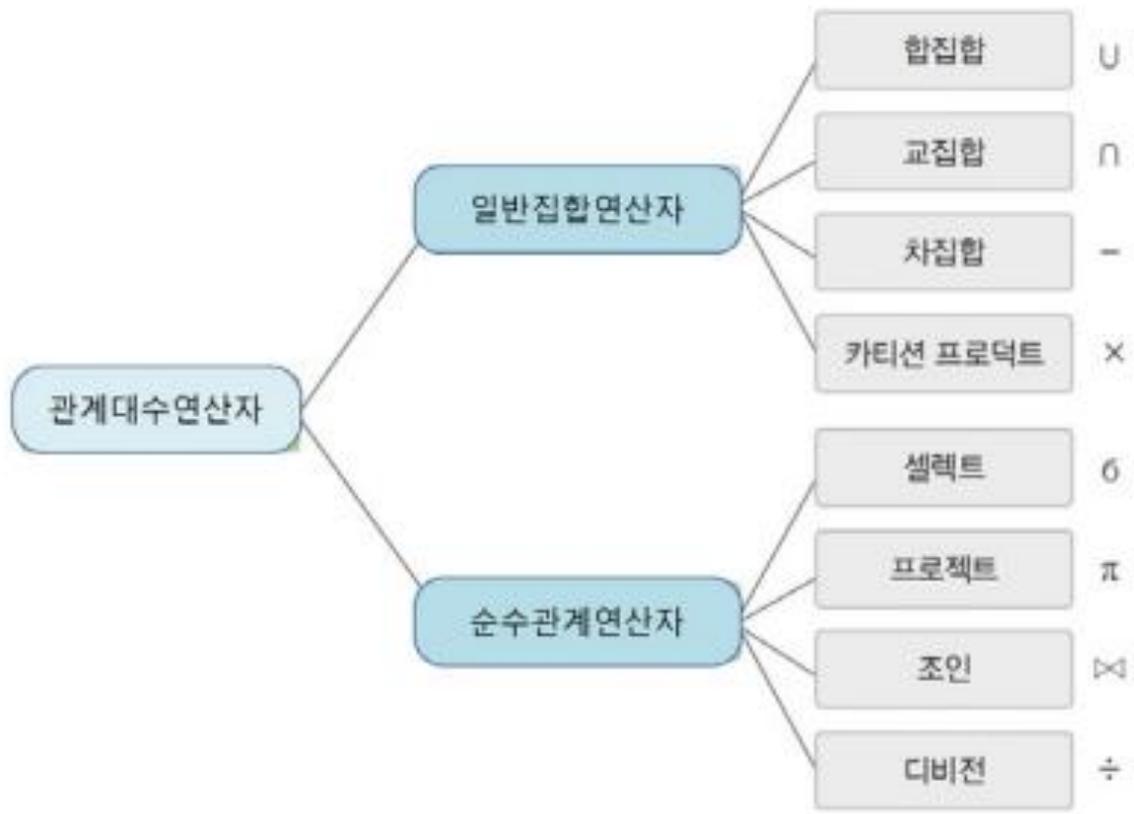


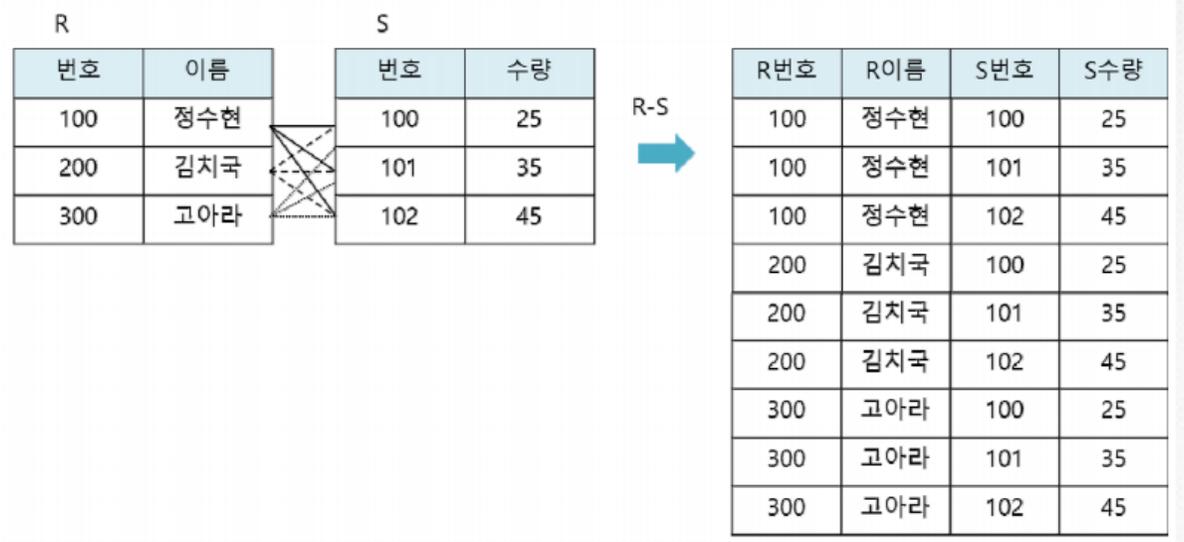
제3과목 데이터베이스 구축

04 논리 데이터베이스 설계 D



관계대수 및 관계해석







선택(Select, σ)

- ① 릴레이션에서 조건에 만족하는 튜플을 검색한다.
- ② $\sigma_{조건식}$ (릴레이션)
- ③ $\sigma_{점수 \geq 90}$ (학생) \Rightarrow 학생 릴레이션에서 점수가 90 이상인 튜플을 검색하시오

[학생]

학번	과목	교수	점수
100	정보통신	정민기	90
100	웹디자인	홍길동	80
103	정보통신	정민기	86
103	웹디자인	홍길동	95
104	사이버안보	정수현	88

\Rightarrow

$\sigma_{점수 \geq 90}$ (학생)

학번	과목	교수	점수
100	정보통신	정민기	90
103	웹디자인	홍길동	95

- ④ $\sigma_{등급='gold'}$ (고객) \Rightarrow 고객 릴레이션에서 등급이 'gold'인 튜플을 검색하시오.





프로젝트(Project, Π)

- ① 릴레이션에서 주어진 속성들의 값으로만 구성된 열을 선택한다.
- ② $\Pi_{(속성리스트)}$ (릴레이션)
- ③ $\Pi_{(과목, 점수)}$ (학생) \Rightarrow 학생 릴레이션에서 과목과 점수를 검색하시오.

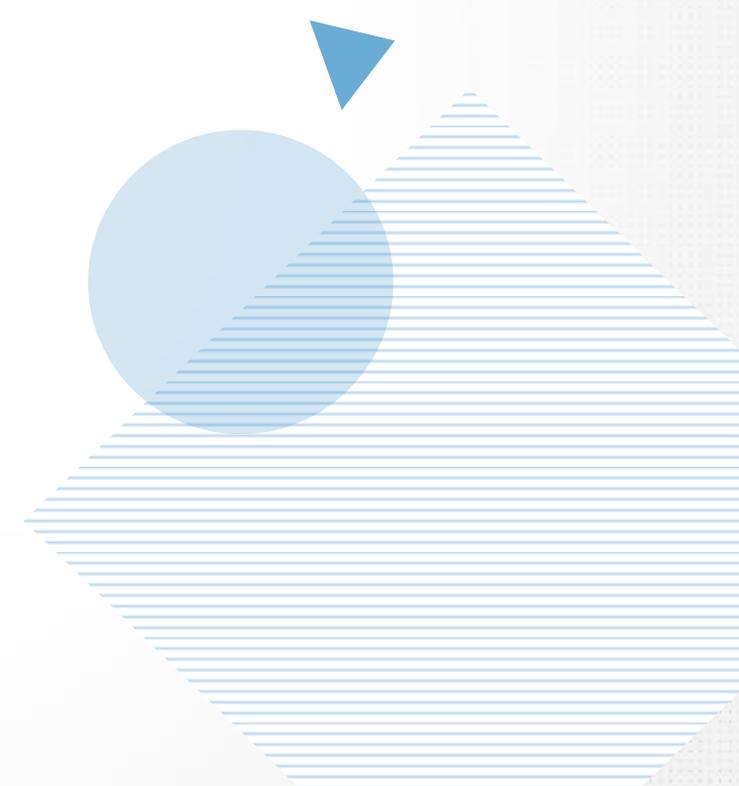
[학생]

학번	과목	교수	점수
100	정보통신	정민기	90
100	웹디자인	홍길동	80
103	정보통신	정민기	86
103	웹디자인	홍길동	95
104	사이버안보	정수현	88

$\Pi_{(과목, 점수)}$ (학생)

과목	점수
정보통신	90
웹디자인	80
정보통신	86
웹디자인	95
사이버안보	88

- ④ $\Pi_{(고객이름, 등급, 적립금)}$ (고객) \Rightarrow 고객 릴레이션에서 고객이름, 등급, 적립금을 검색하시오.
- ⑤ $\pi_{고객이름, 나이}(\sigma_{등급='gold'}(\text{고객}))$ \Rightarrow 고객 릴레이션에서 '등급이 gold'인 고객이름과 나이를 검색하시오.





조인(Join, ⋈)

- ① 공통 속성을 이용하여 2개 이상의 릴레이션을 연결하여 새로운 릴레이션을 생성한다.
- ② 릴레이션1 ⋈_N 릴레이션2
- ③ 사원 ⋈_(부서=부서) 인사

[사원]

사번	성명	부서
100	정민기	A
100	홍길동	B
103	이주호	C
103	정수현	B

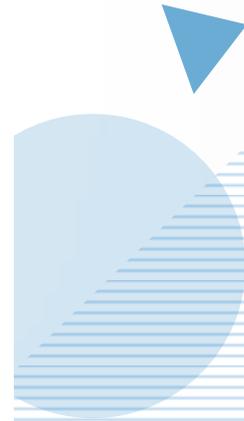
[인사]

부서	부서명
A	기획부
B	총무부
C	인사부

⇒

사원 ⋈_(부서=부서) 인사

사번	성명	부서	부서명
100	정민기	A	기획부
100	홍길동	B	총무부
103	이주호	C	인사부
103	정수현	B	총무부





디비전(Division, ÷)

① B릴레이션의 모든 조건을 만족하는 튜플을 A릴레이션에서 구하는 연산이다.

② 릴레이션1 ÷ 릴레이션2

[고객]

아이디	이름	나이	등급	직업	포인트
app	김현아	20	gold	학생	1000
nul	정소희	25	vip	프리랜서	2500
car	원유인	28	gold	교사	5500
kkk	정지선	22	silver	학생	0

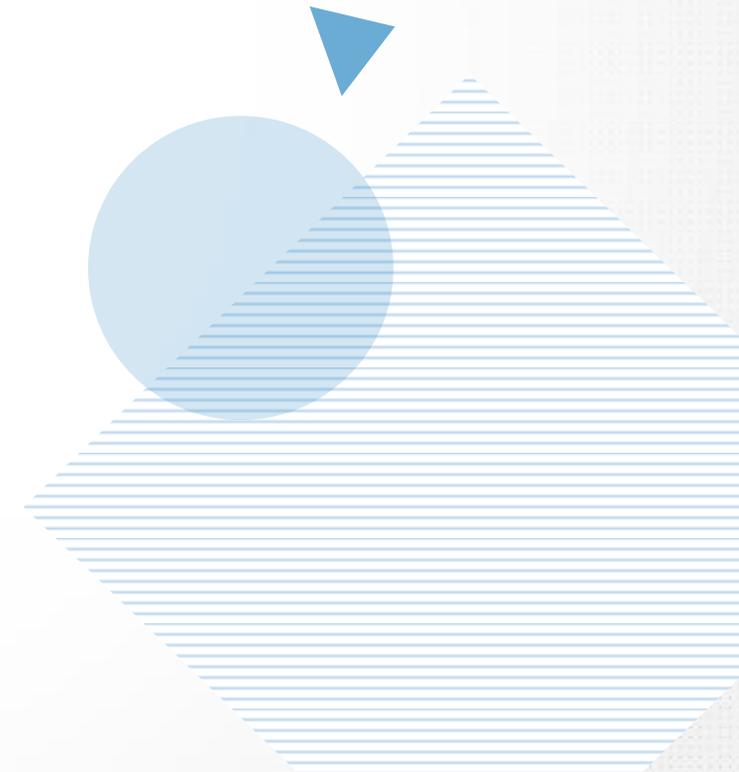
[골드]

등급

gold

③ 고객 ÷ 골드 : 고객 릴레이션에서 등급이 'gold'인 행을 찾은 후 등급속성을 제외하고 나열한다.

아이디	이름	나이	직업	포인트
app	김현아	20	학생	1000
car	원유인	28	교사	5500





[주문내역]

주문고객	제품이름	제조업체
aaa	우동	한양
ccc	파이	미양
bbb	만두	한양
aaa	만두	한양
ccc	만두	한양

[제품]

제품이름
우동
만두

[제조]

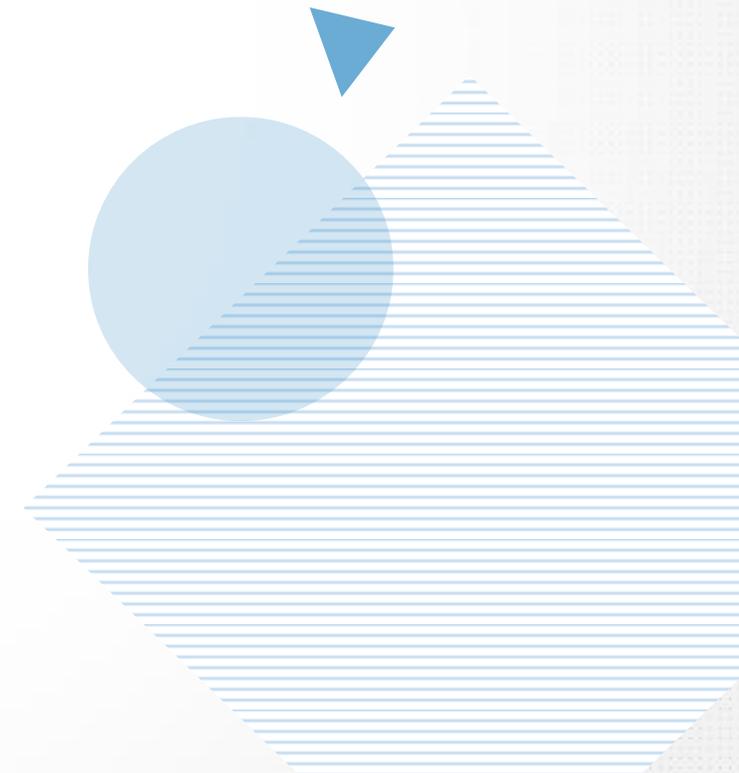
제품이름	제조업체
만두	한양

④ 주문내역 ÷ 제품

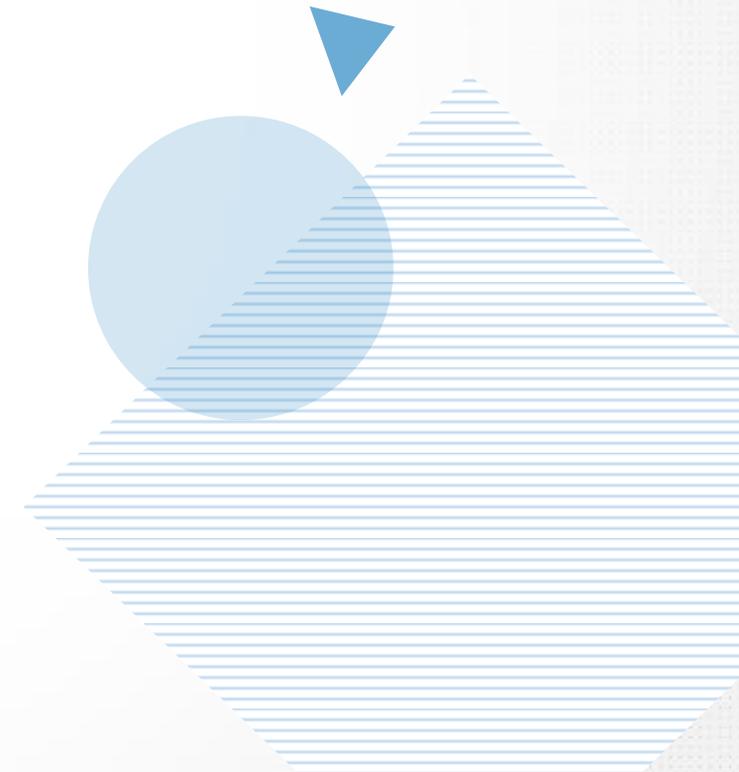
주문내역 ÷ 제품

주문고객	제조업체
aaa	한양

⑤ 주문내역 ÷ 제조



➤ 관계해석(Relational Calculus)

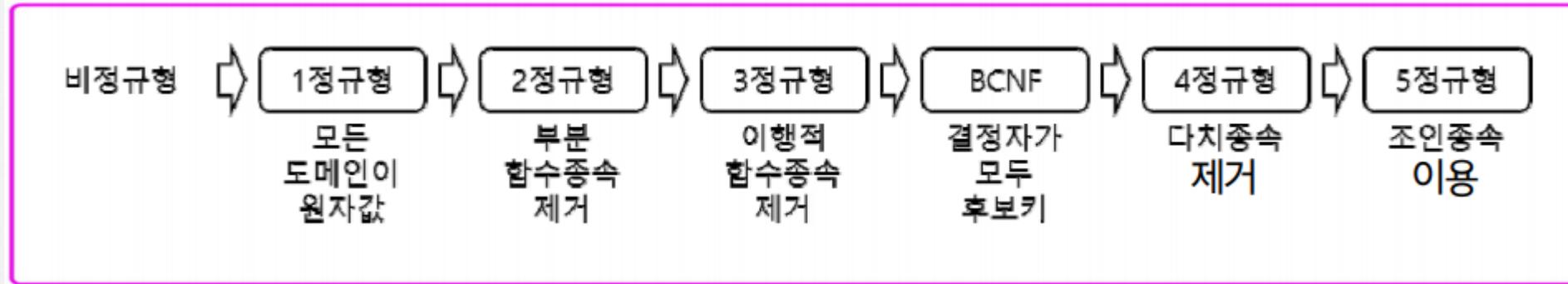


정규화(Normalization)

정규화의 개요

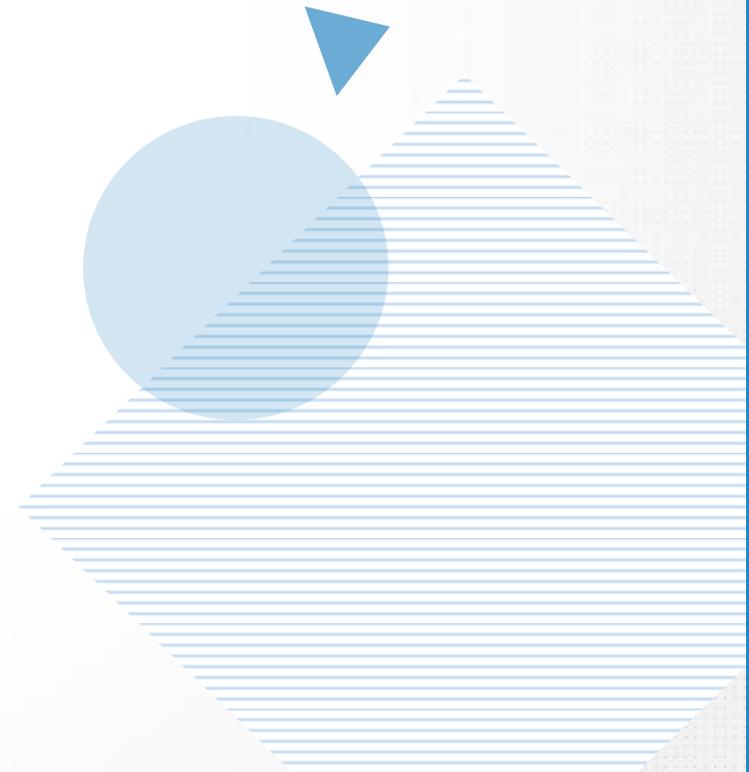
- 삽입이상, 삭제이상, 갱신이상

정규화 과정



반정규화(Denormalization)

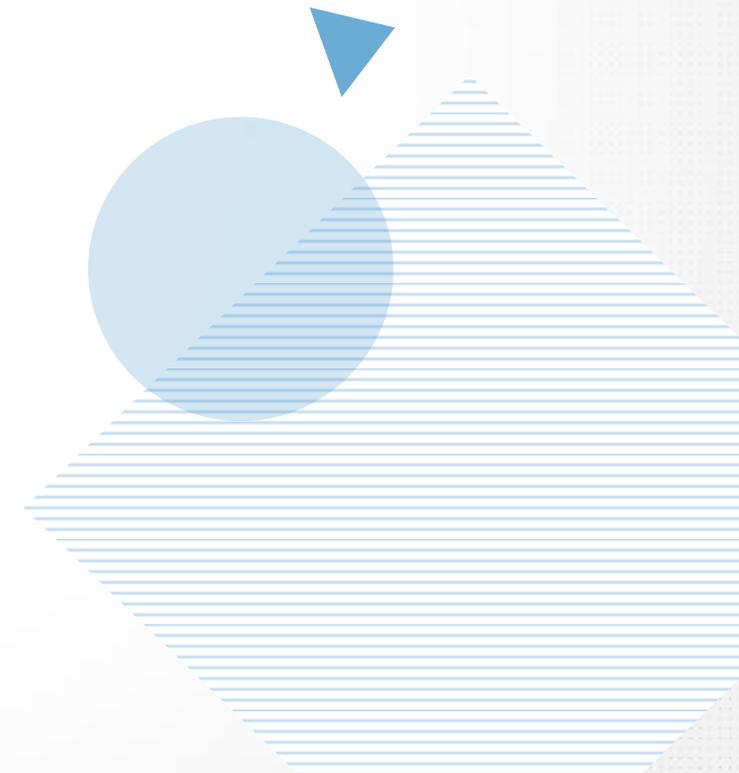
- ▶ 반정규화의 개념
- ▶ 테이블 통합
- ▶ 테이블 분할
- ▶ 중복 테이블 추가
- ▶ 중복 속성 추가



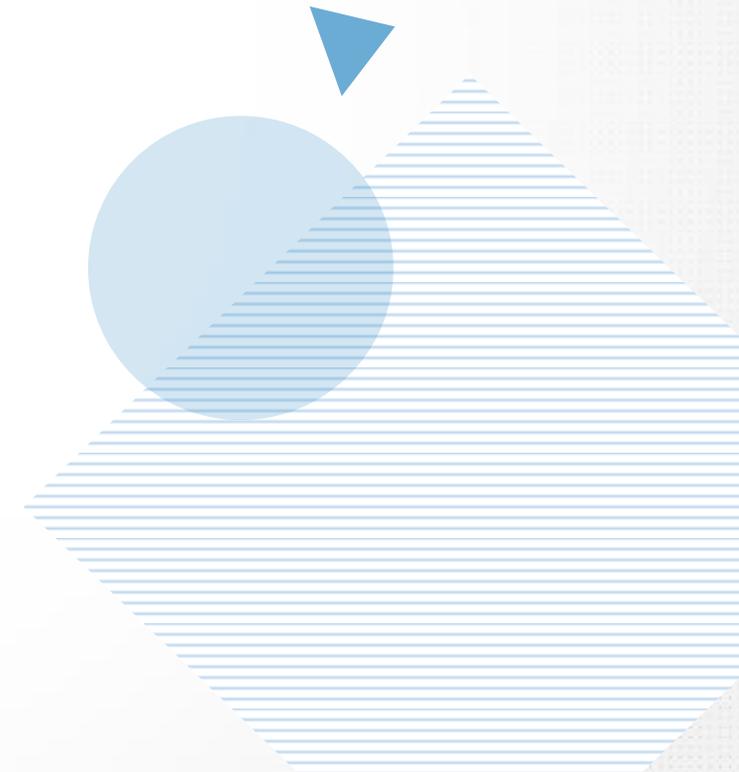
1. 다음 중 시스템의 성능 향상, 개발 및 운영의 편의성 등을 위해 정규화된 데이터 모델을 통합, 중복, 분리하는 과정을 의미하는 용어는?

- ① 통합 정규화
- ② 비 정규화
- ③ 반 정규화
- ④ 강 정규화

정답 3



▶ 시스템 카탈로그(System Catalog)



제3과목 데이터베이스 구축

05 물리 데이터베이스 설계 A



➤ 물리 데이터베이스 설계

- 논리적 데이터베이스를 디스크 등 물리적 저장장치에 저장할 수 있는 물리적 구조의 데이터로 변환하는 과정이다.

➤ 데이터 명명 규칙 파악

- 물리 데이터 모델에 적용해야 하는 규칙으로 물리데이터 모델의 설계 전에 **파악**
- 데이터 표준화 및 논리 데이터베이스 설계의 결과물 등을 통해 파악
- 데이터 명명 규칙을 통해 중복 구축 등을 방지
- 명명 규칙을 파악하려면 도메인과 데이터 사전에 대한 지식이 필요

➤ 시스템 자원 파악

- 데이터베이스 설치에 영향을 미칠 수 있는 물리적인 요소들로, 사전에 미리 파악해야 한다.
- 시스템 자원 : 하드웨어 자원, 운영체제 및 DBMS 버전, DBMS 파라미터(Parameter)정보

➤ 데이터베이스 관리 요소 파악

- 데이터베이스 운영과 관리 요소로, 데이터베이스 시스템의 환경에 따라 달라질 수 있으므로 미리 파악해야 한다.
- 데이터베이스 시스템 조사 분석서 작성
- 데이터베이스 구조, 이중화 구성, 분산 데이터베이스, 접근 제어/접근 통제, DB암호화 등의 범위와 특성 파악

데이터베이스 저장 공간 설계

▶ 테이블(Table)

- 데이터베이스의 가장 기본적인 객체로 행과 열로 구성되어 있다.
- 데이터베이스의 모든 데이터는 테이블에 저장된다.

▶ 일반 테이블

- 현재 사용되는 대부분의 DBMS에서 표준 테이블로 사용되는 테이블 형태

▶ 클러스터드 인덱스 테이블

- 기본키나 인덱스키의 순서에 따라 데이터가 저장되는 테이블

▶ 파티셔닝 테이블

- 대용량의 테이블을 작은 논리적 단위인 파티션으로 나눈 테이블



데이터베이스 저장 공간 설계

➤ 외부 테이블

- 데이터베이스에서 일반 테이블처럼 이용할 수 있는 외부 파일로, 데이터베이스 내에 객체로 존재

➤ 임시 테이블

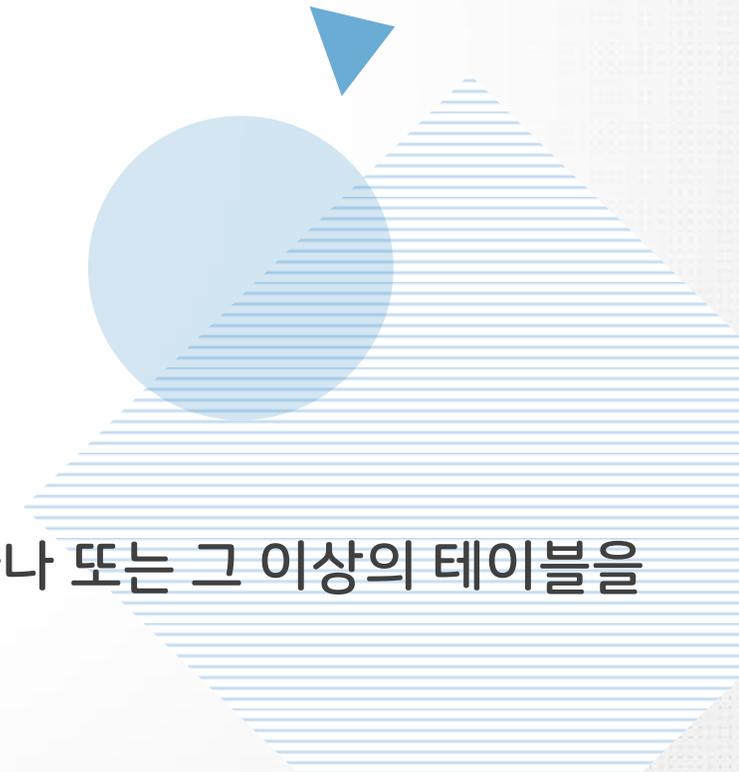
- 트랜잭션이나 세션별로 데이터를 저장하고 처리할 수 있는 테이블

➤ 컬럼

- 테이블의 열을 구성하는 요소로 데이터 타입, 길이 등으로 정의된다.
- 참조 관계인 컬럼들은 데이터 타입과 길이가 일치해야 한다.

➤ 테이블스페이스

- 테이블이 저장되는 논리적인 영역으로, 하나의 테이블스페이스에 하나 또는 그 이상의 테이블을 저장할 수 있다.



▶ 트랜잭션의 정의

- 데이터베이스의 상태를 변환시키는 하나의 논리적 기능을 수행하기 위한 작업의 단위 또는 한꺼번에 모두 수행되어야 할 일련의 연산들을 의미한다.

▶ 트랜잭션의 특성

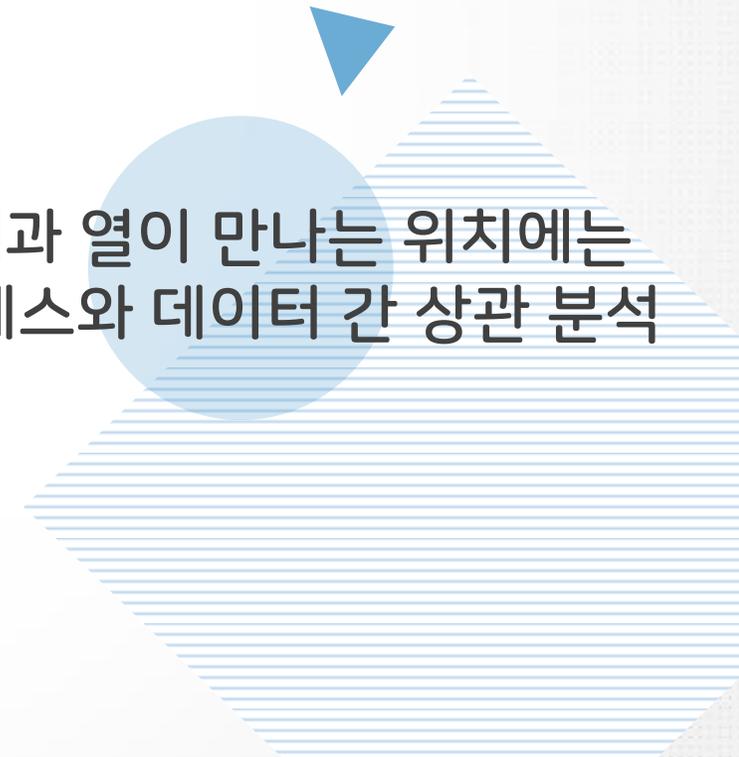
- Atomicity(원자성) : 모두 반영되도록 완료 또는 전혀 반영되지 않도록 복구
- Consistency(일관성) : 트랜잭션의 수행 전과 수행 완료 후의 상태가 같아야 한다.
- Isolation(독립성, 격리성, 순차성) : 하나의 트랜잭션 실행 중에 다른 트랜잭션의 연산이 끼어들 수 없다.
- Durability(영속성, 지속성) : 성공적으로 완료된 트랜잭션의 결과는 영구적으로 반영되어야 한다.

➤ CRUD 분석

- CRUD는 생성(Create), 읽기(Read), 갱신(Udate), 삭제>Delete)의 앞 글자만 모아서 만든 용어
- CRUD 분석은 데이터베이스 테이블에 변화를 주는 트랜잭션의 CRUD 연산에 대해 CRUD 매트릭스를 작성하여 분석하는 것

➤ CRUD 매트릭스

- 2차원 형태의 표로서, 행에는 프로세스를, 열에는 테이블을, 행과 열이 만나는 위치에는 프로세스가 테이블에 발생시키는 변화를 표시하는 업무 프로세스와 데이터 간 상관 분석 표이다.



▶ 트랜잭션 분석

- 테이블에 발생하는 트랜잭션 양을 분석하여 테이블에 저장되는 데이터의 양을 유추하여 DB 용량을 산정하고 DB 구조를 최적화하는 것

▶ 트랜잭션 분석서

- 단위 프로세서와 CRUD 매트릭스를 이용하여 작성
- 구성 요소
 - 단위 프로세스, CRUD 연산, 테이블 명, 컬럼 명, 테이블 참조 횟수, 트랜잭션 수, 발생 주기



▶ 트랜잭션의 특성에 대한 설명으로 옳지 않은 것은?

- ① 원자성은 트랜잭션의 일부만 수행된 상태로 종료될 수 있다는 특성을 의미한다.
- ② 일관성은 시스템의 고정 요소는 트랜잭션 수행 전과 수행 완료 후에 같아야 한다는 특성을 의미한다.
- ③ 격리성은 트랜잭션이 실행될 때마다 다른 트랜잭션의 간섭을 받지 않아야 한다는 성질을 의미한다.
- ④ 지속성은 트랜잭션의 완료 결과가 데이터베이스에 영구히 기억되는 성질을 의미한다.

정답 1

제3과목 데이터베이스 구축

06 물리 데이터베이스 설계 B



▶ 인덱스(Index)의 개념

- 데이터 레코드를 빠르게 접근하기 위해 <키 값, 포인터> 쌍으로 구성되는 데이터 구조
- 데이터가 저장된 물리적 구조와 밀접한 관계가 있다.
- 인덱스를 통해서 파일의 레코드에 대한 액세스를 빠르게 수행할 수 있다.
- 레코드의 삽입과 삭제가 수시로 일어나는 경우에는 인덱스의 개수를 최소화 하는 것이 효율적이다.
- 레코드의 물리적 순서가 인덱스의 엔트리 순서와 일치하도록 구성되는 인덱스를 클러스터드(Clustered) 인덱스라고 한다.

▶ 트리 기반 인덱스

- 인덱스를 저장하는 블록들이 트리 구조를 이루고 있는 것
- B 트리 인덱스 : 일반적으로 사용되는 인덱스 방식
- B+ 트리 인덱스 : B 트리의 변형, 인덱스 세트(Index Set), 순차 세트

▶ 비트맵 인덱스

- 인덱스 컬럼의 데이터를 Bit 값인 0 또는 1로 변환하여 인덱스 키로 사용하는 방법

▶ 함수 기반 인덱스

- 컬럼의 값 대신 컬럼에 특정 함수나 수식을 적용하여 산출된 값을 사용하는 것

인덱스 설계

▶ 비트맵 조인 인덱스

- 다수의 조인된 객체로 구성된 인덱스

▶ 도메인 인덱스

- 개발자가 필요한 인덱스를 직접 만들어 사용하는 것



▶ 인덱스 설계

- 분명하게 드러난 컬럼에 대해 기본적인 인덱스를 먼저 저장한 후 개발 단계에서 필요한 인덱스의 설계를 반복적으로 진행

▶ 인덱스 대상 테이블 선정 기준

- MULTI BLOCK READ 수에 따라 판단

▶ 인덱스 대상 컬럼 선정 기준

- 인덱스 컬럼의 분포도가 10~15% 이내인 컬럼

▶ 인덱스 설계 시 고려사항

- 새로 추가되는 인덱스는 기존 인덱스 경로에 영향을 미칠 수 있다.



뷰(View)의 설계

▶ 뷰(View)의 개요

- 사용자에게 접근이 허용된 자료만을 제한적으로 보여주기 위해 하나 이상의 기본 테이블로부터 유도된 가상 테이블이다.

▶ 뷰(View)의 특징

- 물리적으로 구형되어 있지 않다. 데이터의 논리적 독립성을 제공할 수 있다.

▶ 뷰(View)의 장점

- 사용자의 데이터 관리를 간단하게 해준다. 접근 제어를 통한 자동 보안이 제공된다.

▶ 뷰(View)의 단점

- 뷰로 구성된 내용에 대한 삽입, 삭제, 갱신 연산에 제약이 따른다

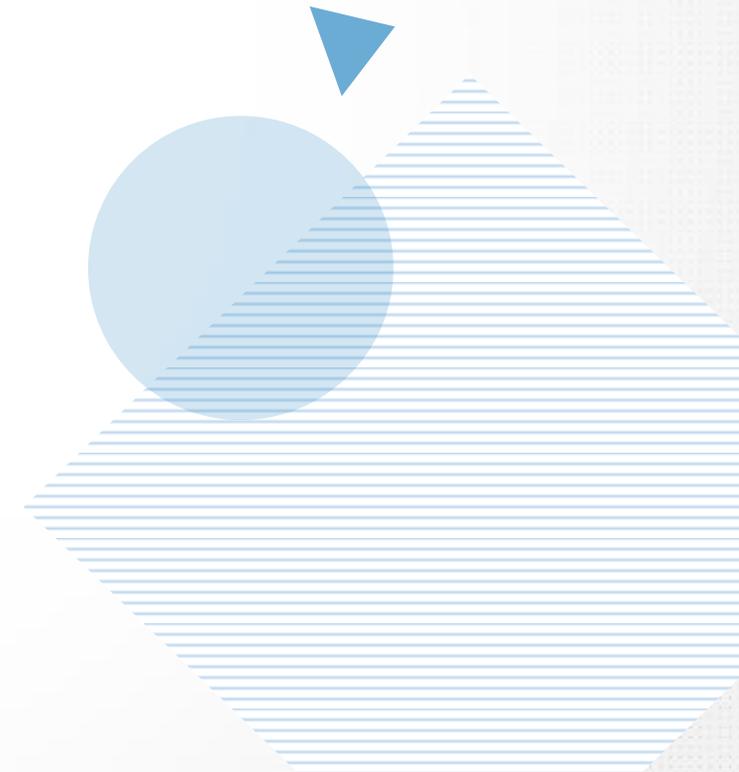
③ 뷰(View)의 설계

➤ 뷰(View) 설계 순서

- 대상 테이블을 선정한다. - 대상 컬럼을 선정한다. - 정의서를 작성한다.

➤ 뷰 설계 시 고려 사항

- 반복적으로 조인을 설정하여 사용하거나 동일한 조건절을 사용하는 테이블을 뷰로 생성한다.



클러스터 설계

▶ 클러스터(Cluster)의 개요

- 데이터 저장 시 데이터 액세스 효율을 향상시키기 위해 동일한 성격의 데이터를 동일한 데이터 블록에 저장하는 물리적 저장 방법이다.

▶ 클러스터(Cluster)의 특징

- 데이터 조회 속도는 향상시키지만 데이터 입력, 수정, 삭제에 대한 성능은 저하시킨다.
- 파티셔닝된 테이블에는 클러스터링을 할 수 없다.

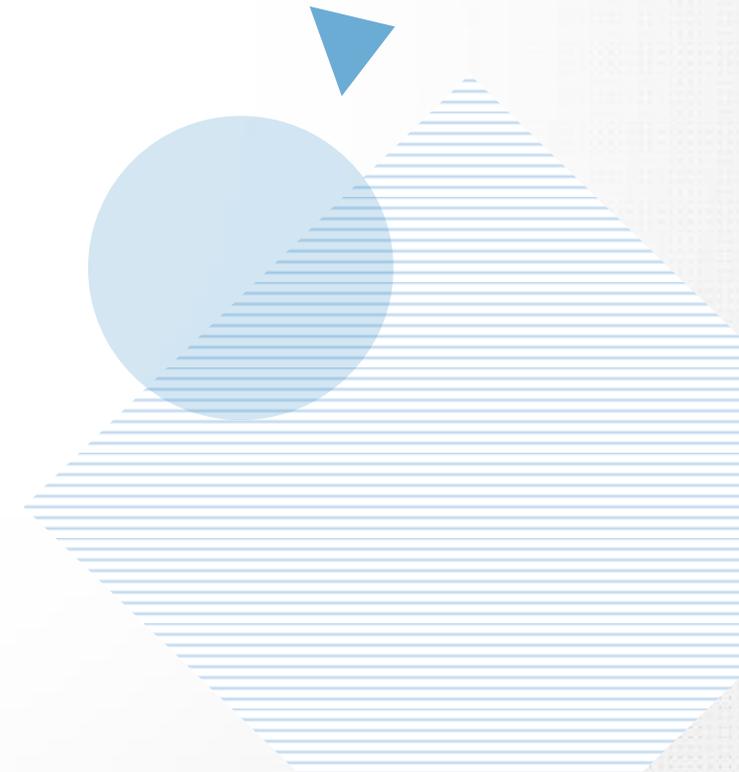
▶ 클러스터 대상 테이블

- 분포도가 넓은 테이블, 대량의 범위를 자주 조회하는 테이블, 입력/수정/삭제가 자주 발생하지 않는 테이블, 자주 조인되어 사용되는 테이블

1. 다음 중 클러스터링을 적용하기에 적당하지 않은 테이블은?

- ① 분포도가 좁은 테이블
- ② 수정이 거의 발생하지 않는 테이블
- ③ 자주 조인되어 사용되는 테이블
- ④ 대량의 범위를 자주 조회하는 테이블

정답1



제3과목 데이터베이스 구축

07 물리 데이터베이스 설계 C



▶ 파티션(Partition)의 개요

- 대용량의 테이블이나 인덱스를 작은 논리적 단위인 파티션으로 나누는 것이다.

▶ 파티션의 장점

- 파티션별로 데이터가 분산되어 저장되므로 디스크의 성능이 향상된다.
- 파티션별로 백업 및 복구를 수행하므로 속도가 빠르다.
- 시스템 장애 시 데이터 손상 정도를 최소화할 수 있다.

▶ 파티션의 단점

- 세심한 관리가 요구된다, 테이블간 조인에 대한 비용이 증가한다.

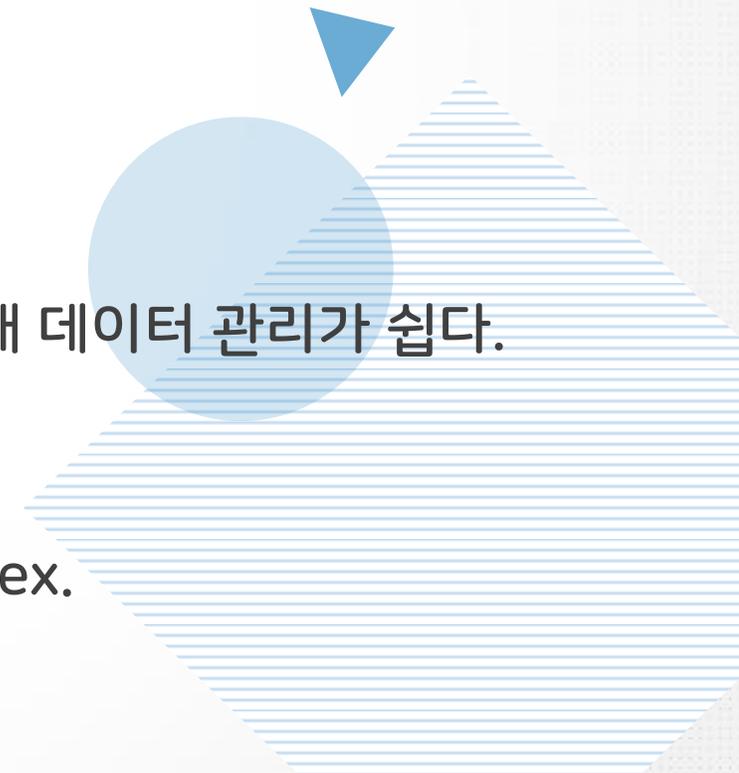


▶ 파티션(Partition)의 종류

- 범위 분할(Range Partitioning), 해시분할(Hash Partitioning)
- 조합 분할(Composite Partitioning)

▶ 인덱스 파티션

- 파티션된 테이블의 종속 여부에 따라
 - Local Partitioned Index. Global Partitioned Index.
 - Local Partitioned Index 가 Global Partitioned Index 에 비해 데이터 관리가 쉽다.
- 인덱스 파티션키 컬럼의 위치에 따라
 - Prefixed Partitioned Index. Non-Prefixed Partitioned Index.



데이터베이스 용량 설계

▶ 데이터베이스 용량 설계

- 데이터가 저장될 공간을 정의하는 것이다

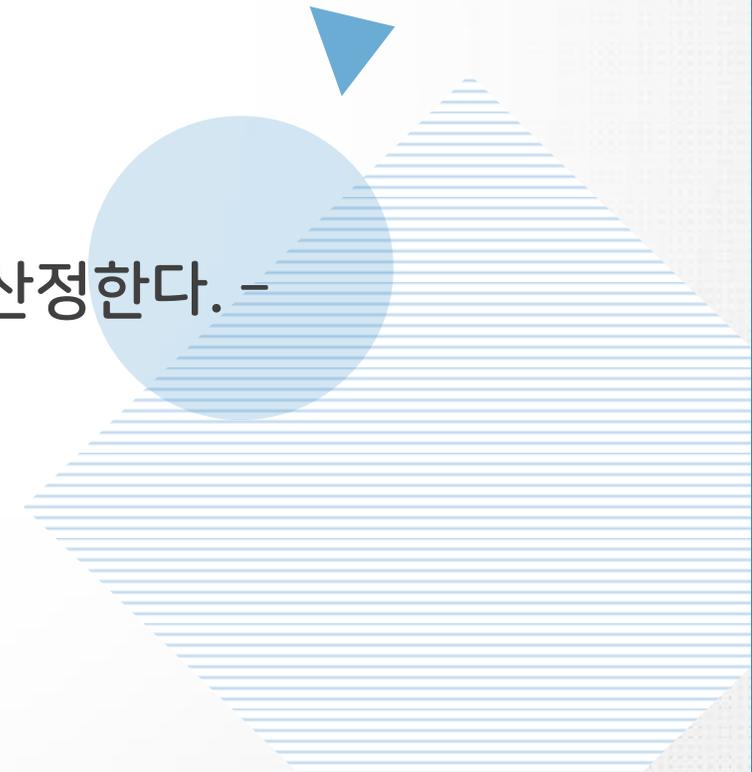
▶ 데이터베이스 용량 설계의 목적

- 용량을 정확히 산정하여 디스크의 저장 공간을 효과적으로 사용하고 확장성 및 가용성을 높인다.

▶ 데이터베이스 용량 분석 절차

- 기초 자료를 수집하여 용량을 분석한다. - 오브젝트별 용량을 산정한다. -

-



분산 데이터베이스 설계

➤ 분산 데이터베이스 정의

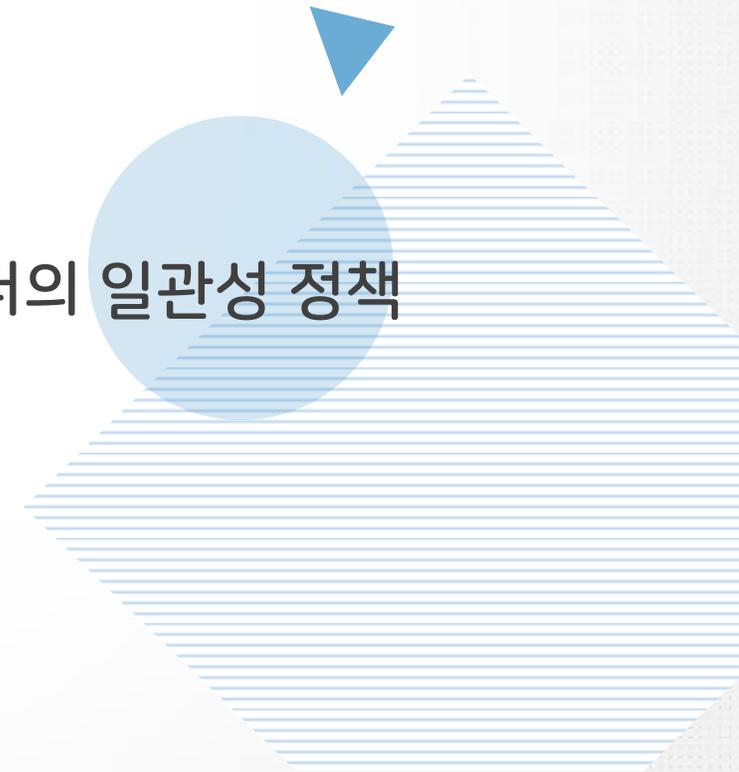
- 논리적으로는 하나의 시스템에 속하지만 물리적으로는 네트워크를 통해 연결된 여러 개의 컴퓨터 사이트에 분산되어 있는 데이터베이스이다.

➤ 분산 데이터베이스의 구성 요소

- 분산 처리기, 분산 데이터베이스, 통신 네트워크

➤ 분산 데이터베이스 설계 시 고려 사항

- 작업부하의 노드별 분산 정책, 지역의 자치성 보장 정책, 데이터의 일관성 정책



분산 데이터베이스 설계

➤ 분산 데이터베이스의 목표

- 위치 투명성(Location Transparency), 중복 투명성(Rrplication Transparency)
- 병행 투명성(Concurrency Transparency), 장애 투명성(Failure Transparency)

➤ 분산 데이터베이스 장·단점

- 장점 : 지역 자치성이 높다, 시스템 성능이 향상된다, 효율성과 융통성이 높다.
- 단점 : 소프트웨어 개발 비용이 증가한다, 잠재적 오류가 증가한다.



분산 데이터베이스 설계

➤ 분산 데이터베이스 설계

- 애플리케이션이나 사용자가 분산되어 저장된 데이터에 접근하게 하는 것이 목적

➤ 테이블 위치 분산

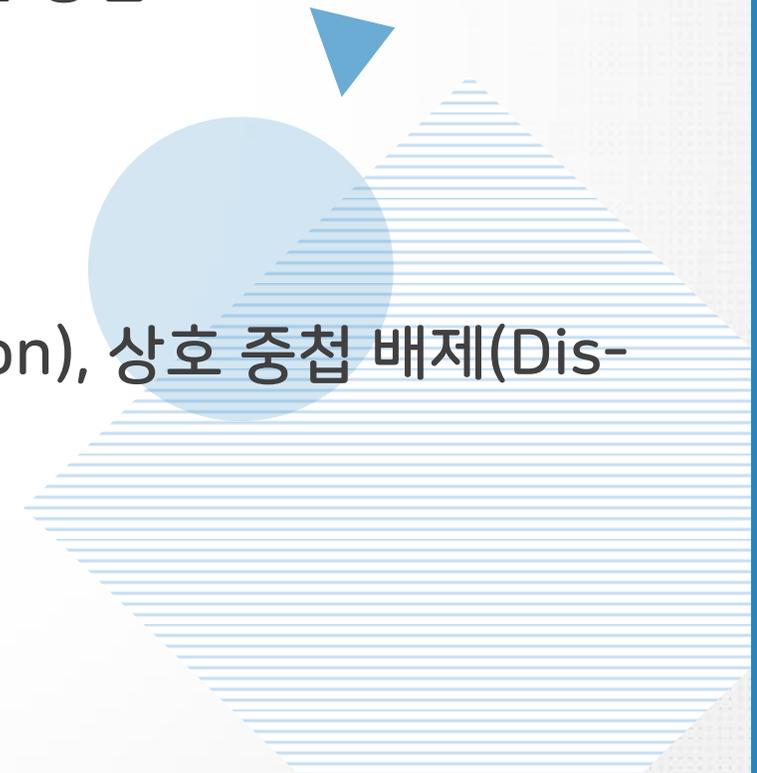
- 데이터베이스의 테이블을 각기 다른 서버에 분산시켜 배치하는 방법

➤ 분할(Fragment)

- 테이블의 데이터를 분할하여 분산시키는 것
- 분할 규칙 : 완전성(Completeness), 재구성(Reconstruction), 상호 중첩 배제(Disjointness)

➤ 할당(Allocation)

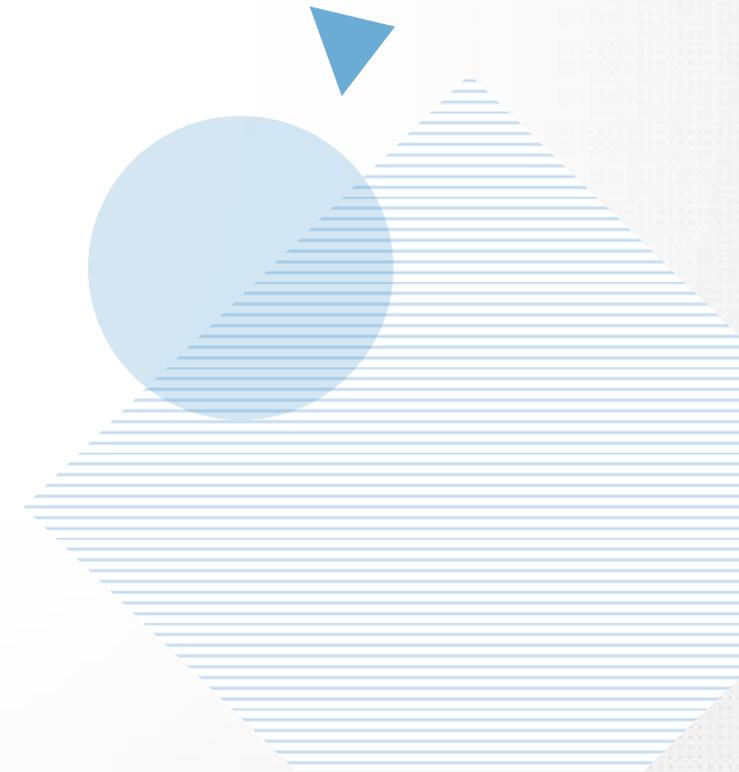
- 동일한 분할을 여러 개의 서버에 생성하는 분산 방법



1. 분산 데이터베이스 시스템이 사용자에게 제공하는 네가지 유형의 투명성에 속하지 않는 것은?

- ① 위치 투명성
- ② 복제 투명성
- ③ 수행 투명성
- ④ 병행 투명성

정답 3



제3과목 데이터베이스 구축

08 물리 데이터베이스 설계 D



🔄 데이터베이스 이중화/서버 클러스터링

- 데이터베이스 이중화(Database Replication)
 - 동일한 데이터베이스를 복제하여 관리하는 것이다.
- 데이터베이스 이중화의 분류(변경 내용의 전달 방식에 따라)
 - Eager 기법 , Lazy 기법
- 데이터베이스 이중화 구성 방법
 - 활동-대기(Active-Standby) 방법, 활동-활동(Active-Active) 방법
- 클러스터링(Clustering)
 - 두 대 이상의 서버를 하나의 서버처럼 운영하는 기술



🔒 데이터베이스 보안/암호화

➤ 데이터베이스 보안의 개요

- 데이터베이스의 권한이 없는 사용자가 액세스하는 것을 금지하기 위해 사용되는 기술

➤ 암호화(Encryption)

- 개인키 암호 방식(Private Key Encryption) = 비밀키 암호 방식
 - 동일한 키로 데이터를 암호화하고 복호화한다.
- 공개키 암호 방식(Public Key Encryption)
 - 서로 다른 키로 데이터를 암호화하고 복호화한다.

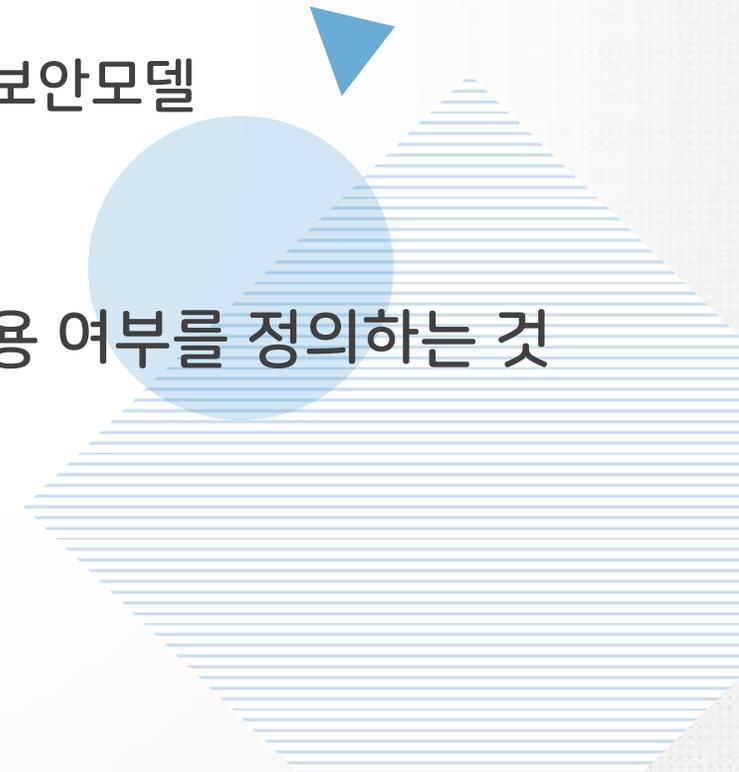


➤ 접근통제

- 객체와 이를 사용하려는 주체 사이의 정보 흐름을 제한하는 것
- 임의 접근통제(DAC)
- 강제 접근통제(MAC)
- 접근통제의 3요소 : 접근통제 정책, 접근통제 메커니즘, 접근통제 보안모델

➤ 접근통제 정책

- 어떤 주체가 언제, 어디서, 어떤 객체에게 어떤 행위에 대한 허용 여부를 정의하는 것
- 신분기반 정책, 규칙기반 정책, 역할기반 정책



데이터베이스 보안-접근통제

➤ 접근통제 매커니즘

- 정의된 접근통제 정책을 구현하는 기술적인 방법.
- 접근통제 목록, 능력 리스트, 보안 등급, 비밀번호, 암호화

➤ 접근통제 보안 모델

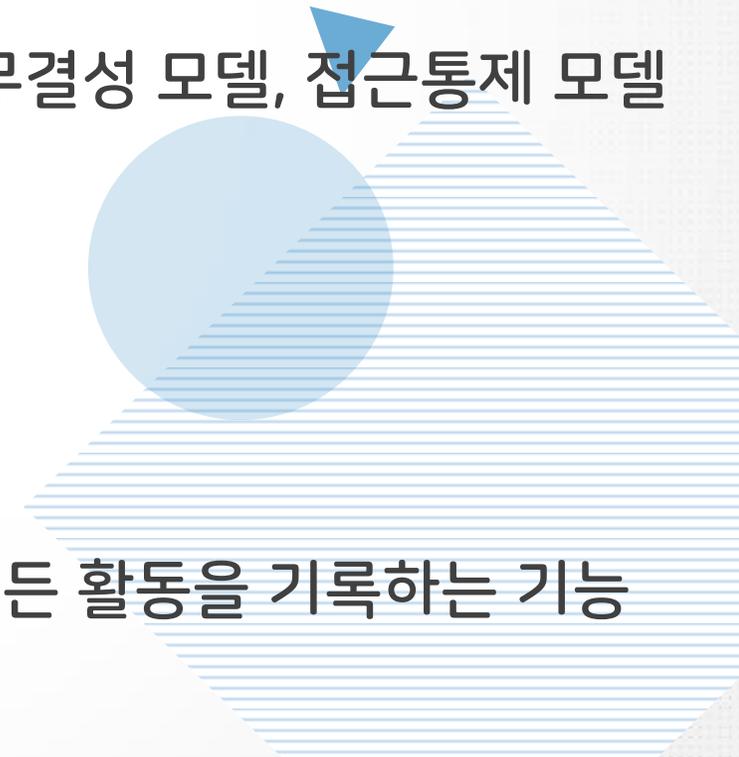
- 보안 정책을 구현하기 위한 정형화된 모델이다. 기밀성 모델, 무결성 모델, 접근통제 모델

➤ 접근통제 조건

- 값 종속 통제, 다중 사용자 통제, 컨텍스트 기반 통제

➤ 감사 추적

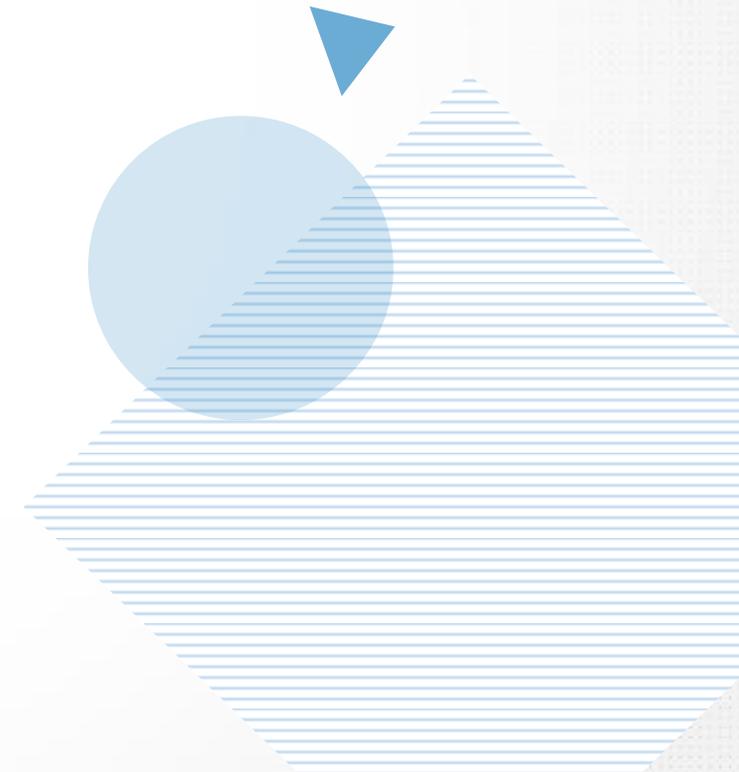
- 사용자나 애플리케이션이 데이터베이스에 접근하여 수행한 모든 활동을 기록하는 기능



1. 다음 중 접근통제의 3요소가 아닌것은 ?

- ① 접근통제 정책
- ② 접근통제 보안 모델
- ③ 접근통제 조건
- ④ 접근통제 메커니즘

정답 3



제3과목 데이터베이스 구축

09 물리 데이터베이스 설계 E



데이터베이스 백업

▶ 데이터베이스 백업

- 장애에 대비하여 데이터베이스에 저장된 데이터를 보호하고 복구하기 위한 작업이다.

▶ 데이터베이스 장애 유형

- 사용자 실수, 미디어 장애, 구문 장애, 사용자 프로세스 장애, 인스턴스 장애

▶ 데이터베이스 복구 알고리즘

- NO-UNDO/REDO, UNDO/NO-REDO, UNDO/REDO, NO-UNDO/NO-REDO

▶ 백업 종류

- 물리 백업, 논리 백업



➤ 스토리지(Storage)의 개요

- 대용량의 데이터를 저장하기 위해 서버와 저장장치를 연결하는 기술

➤ DAS(Direct Attached Storage)

- 서버와 저장장치를 전용 케이블로 직접 연결하는 방식

➤ NAS(Network Attached Storage)

- 서버와 저장장치를 네트워크를 통해 연결하는 방식

➤ SAN(Storage Area Network)

- 서버와 저장장치를 연결하는 전용 네트워크를 별도로 구성하는 방식



논리 데이터 모델의 물리 데이터 모델 변환

▶ 테이블(Table)

- 데이터를 저장하는 데이터베이스의 가장 기본적인 오브젝트이다.
- 로우(Row), 컬럼(Column), 기본키(Primary key), 외래키(Foreign key)

▶ 엔티티(Entity)를 테이블로 변환

- 논리 데이터 모델에서 정의된 엔티티를 물리 데이터 모델의 테이블로 변환하는 것

▶ 슈퍼타입/서브타입을 테이블로 변환

- 물리 데이터 모델을 설계할 때는 슈퍼타입/서브타입을 테이블로 변환해야 한다.
- 슈퍼타입 기준 테이블 변환, 서브타입 기준 테이블 변환, 개별타입 기준 테이블 변환

논리 데이터 모델의 물리 데이터 모델 변환

➤ 속성을 컬럼으로 변환

- 일반 속성 변환, Primary UID를 기본키로 변환
- Primary UID(관계의 UID Bar)를 기본키로 변환
- Secondary(Alternate) UID를 유니크키로 변환

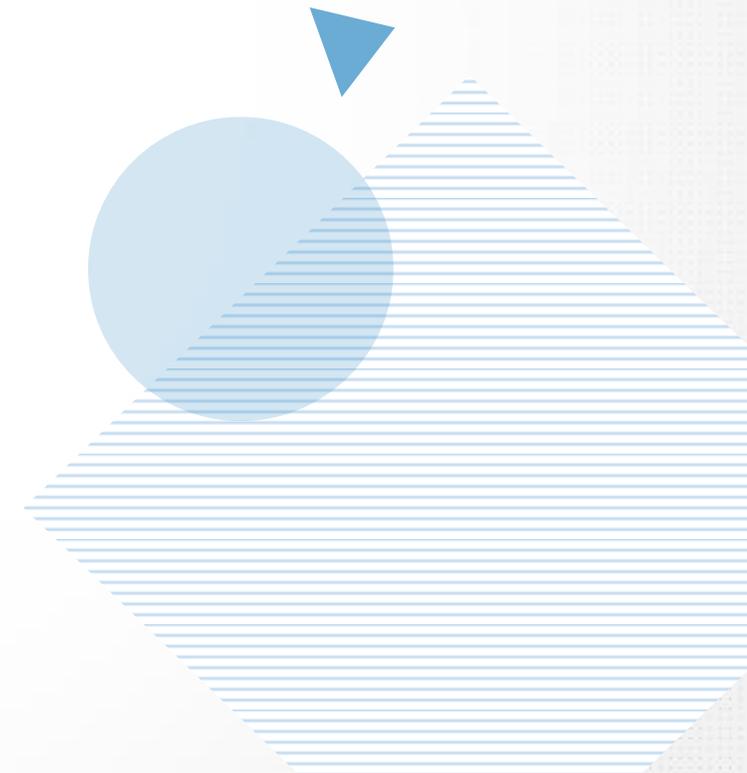
➤ 관계를 외래키로 변환

- 1:1 관계, 1:M 관계, N:M 관계

➤ 관리 목적의 테이블/컬럼 추가

➤ 데이터 타입 선택

- 문자 타입, 숫자 타입, 날짜 타입



물리 데이터 모델 품질 검토

물리 데이터 모델 품질 검토

- 물리 데이터 모델 품질 검토의 목적은 데이터베이스의 성능 향상과 오류 예방이다.
- 물리 데이터 모델을 검토하려면 모든 이해관계자가 동의하는 검토 기준이 필요하다.

물리 데이터 모델 품질 기준

- 정확성, 완전성, 준거성, 최신성, 일관성, 활용성이다.
- 조직 혹은 업무 상황에 따라 가감하거나 변형하여 사용한다.

물리 데이터 모델 품질 검토 항목

- 물리 데이터 모델의 특성을 반영한 품질 기준을 작성한 후 이를 기반으로 작성한다.

물리 데이터 모델의 품질 검토 순서



1. 다음 중 논리 데이터베이스에서 설계한 객체들을 물리 데이터베이스로 변환하는 과정에 대한 설명으로 틀린것은 ?

- ① 엔티티는 테이블로 변환한다.
- ② 속성은 칼럼으로 변환한다.
- ③ 주 시별자는 외래키로 변환한다.
- ④ 테이블명은 영문으로 지정한다.

정답 3

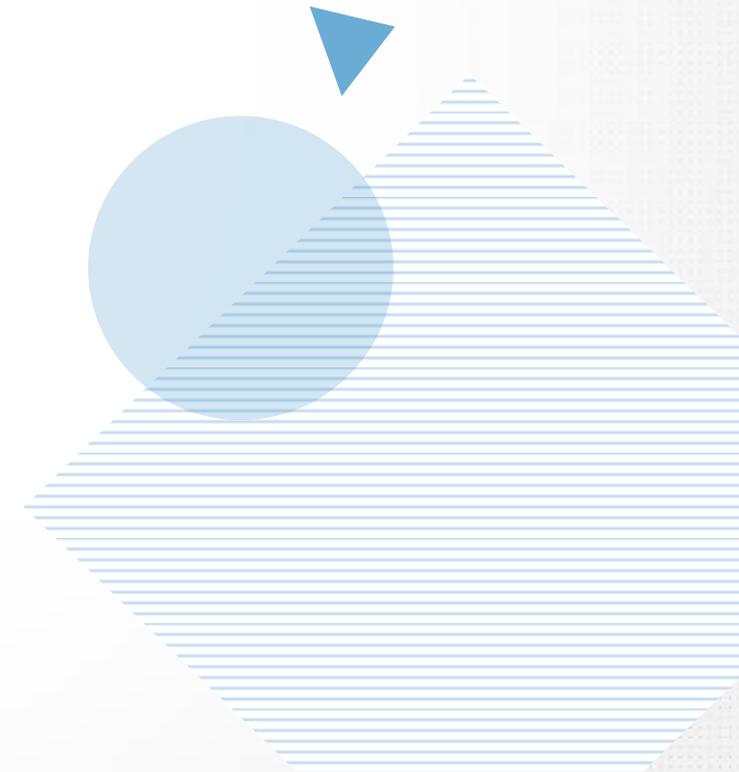


제3과목 데이터베이스 구축

10 SQL 응용 A

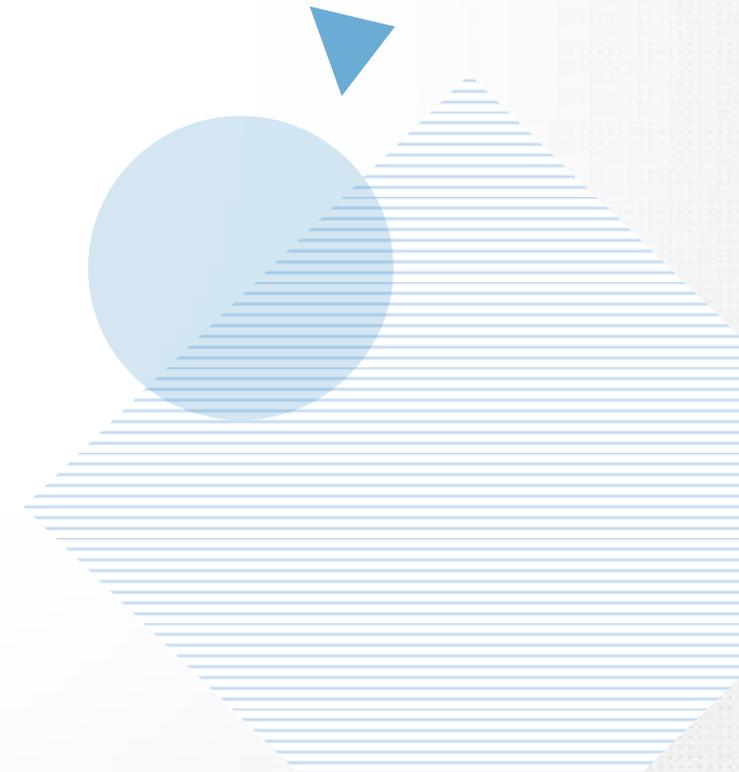


- SQL(Structured Query Language)의 개요
- SQL의 분류

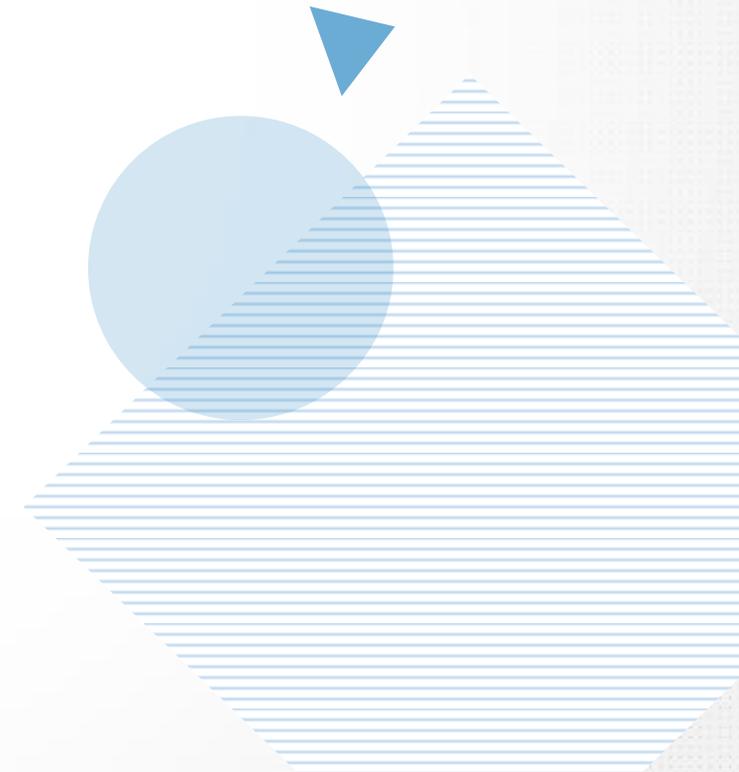


➤ DDL(Data Define Language, 데이터 정의어)의 개념

- CREATE SCHEMA
- CREATE DOMAIN
- CREATE TABLE
- CREATE VIEW
- CREATE INDEX
- ALTER TABLE
- DROP



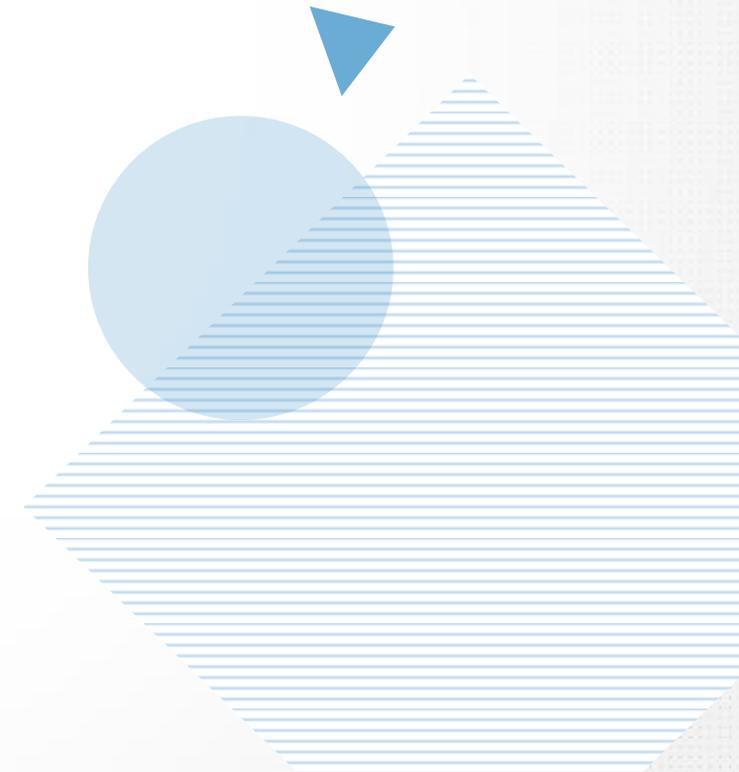
- DCL(Data Control Language, 데이터 제어어)의 개념
 - GRANT / REVOKE
 - COMMIT
 - ROLLBACK
 - SAVEPOINT



1. SQL언어의 데이터 제어어에 해당되는 것은?

- ① Select
- ② Insert
- ③ Update
- ④ Grant

정답 4

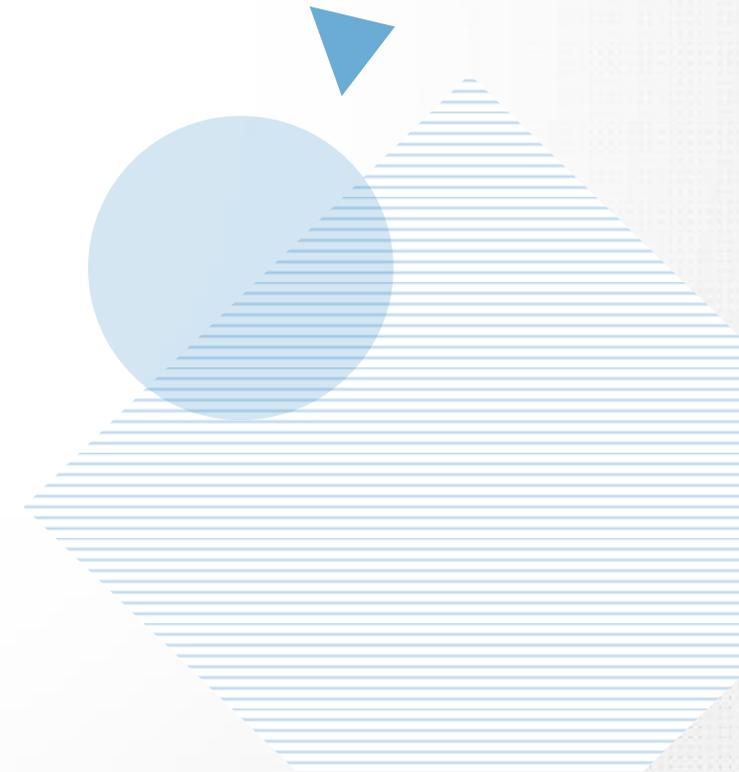


제3과목 데이터베이스 구축

11 SQL 응용 B



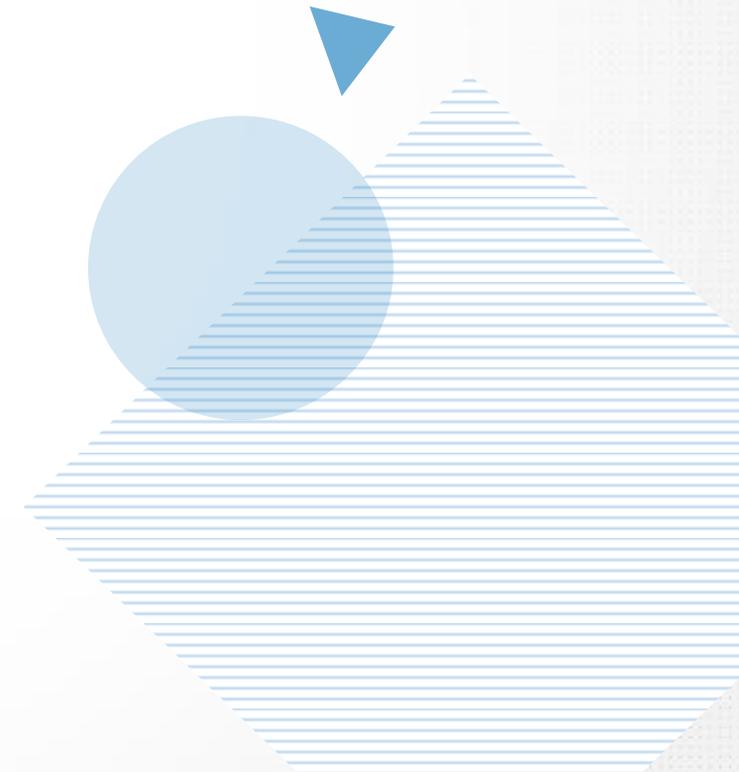
- DML(Data Manipulation Language, 데이터 조작용어)의 개념
 - 삽입문(INSERT INTO~)
 - 삭제문(DELETE FROM~)
 - 갱신문(UPDATE~ SET~)
 - 검색 select ~ from



➤ INSERT

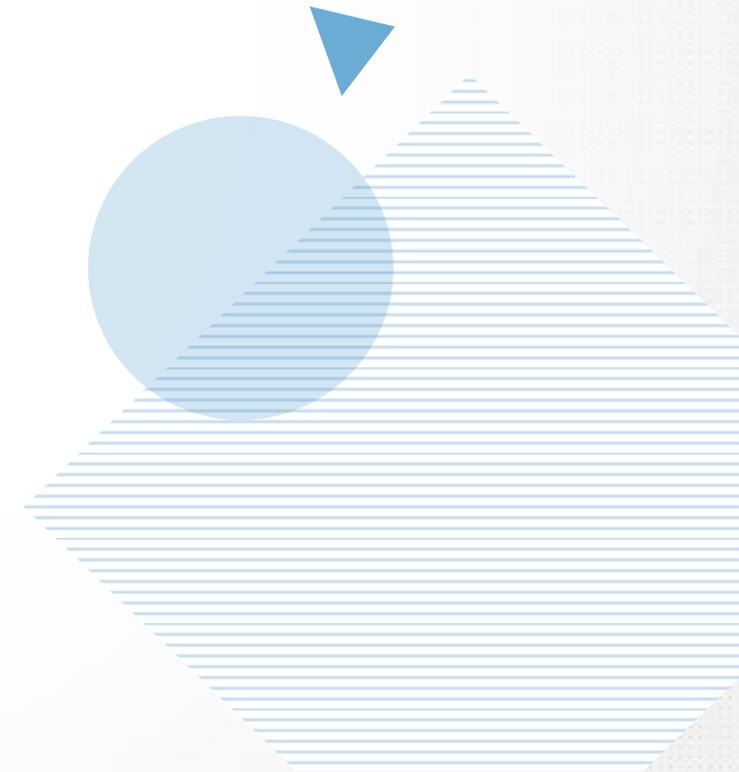
- 새로운 튜플을 삽입하는 명령이다.
 - INSERT INTO 테이블명(속성1, 속성2...) VALUE(데이터1, 데이터2..)
 - INSERT INTO 학생(번호, 이름, 학년, 학과) VALUES('010', '정수현', 2, '사이버안보')
 - 학생테이블의 번호, 이름, 학년, 학과 속성에 '010', '정수현', 2, '사이버안보'를 삽입하십시오.
- 

- INSERT INTO 학생 VALUES ('홍길동','컴퓨터','서울')
- INSERT INTO 학생 (이름,수강료) VALUES (홍길동,120)
 - INSERT INTO 수강정보(이름, 과목, 수강료)
 - SELECT 이름, 과목, 수강료
 - FROM 학생
 - WHERE 주소='서울'



➤ DELETE

- 테이블의 특정 튜플을 삭제하는 명령이다.
- DELETE FROM 테이블명 WHERE 조건
- DELETE FROM 학생 WHERE 번호='101'
- DELETE FROM 학생
 - WHERE 과목='컴퓨터'



➤ UPDATE

- 테이블의 특정 튜플의 내용을 변경하고자 할 때 사용한다.
- UPDATE 테이블명
SET 속성명=데이터
WHERE 조건;
- UPDATE 학생 SET 이름='홍길동' WHERE 번호='101'

```
UPDATE 학생 SET 수강과목='컴퓨터' WHERE 이름='홍길동'
```

```
UPDATE 학생 SET 수강료=수강료+10000 WHERE 수강과목='컴퓨터'
```

➤ SELECT 명령 - 검색

- SELECT [DISTINCT] 속성명

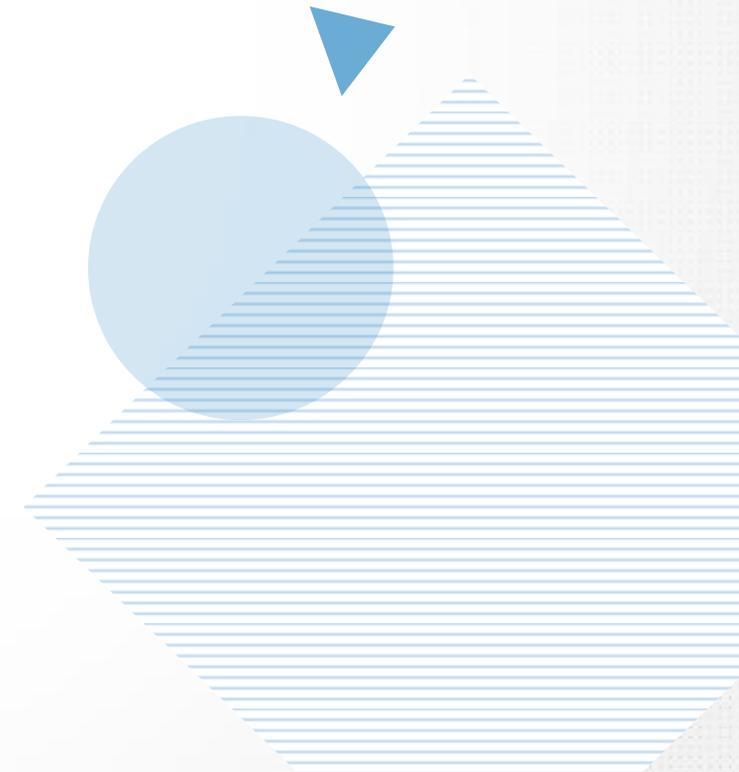
FROM 테이블명

WHERE 조건

ORDER BY 정렬할 속성명 [ASC] 또는 [DESC]

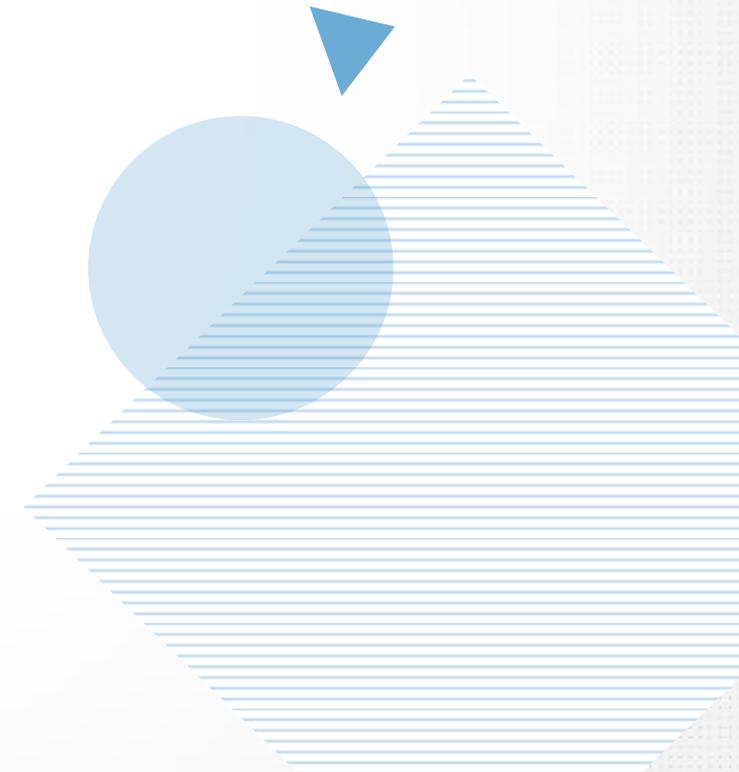
GROUP BY 그룹화 속성명

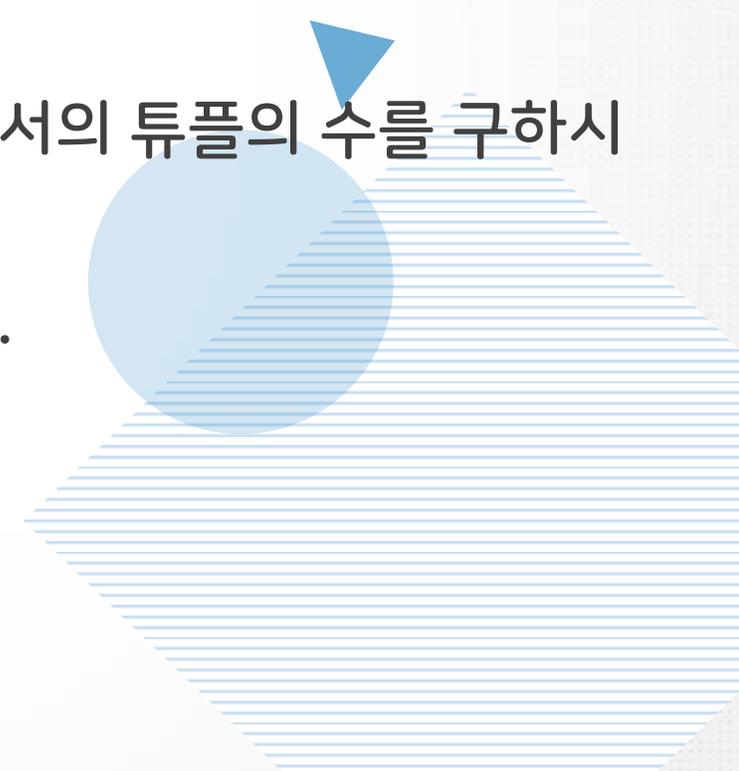
HAVING 그룹조건;



- <사원>테이블의 모든 튜플을 검색하시오.
 - <사원>테이블에서 주소속성만 나타내는데 같은 주소는 한번만 출력하시오.
 - <사원>테이블에서 급여에 10을 더한 값을 '○○부서의 ○○○의 급여는 ○○○입니다.' 형태로 출력하시오.
 - <사원>테이블에서 부서가 '편집'부인 모든 튜플을 검색하시오.
 - <사원>테이블에서 부서가 '편집'부이면서 주소가 '후월동'인 튜플을 검색하시오
- 

- <사원>테이블에서 부서가 '편집'이거나 '기획'인 튜플을 검색하시오.
- <사원>테이블에서 성이 '홍'인 사람의 튜플을 검색하시오.
- <사원>테이블에서 생일이 '1970-01-01'~'1980-12-31'사이인 튜플을 검색하시오.
- <사원>테이블에서 급여가 210~220인 튜플을 검색하시오.
- <사원>테이블에서 주소가 NULL인 튜플을 검색하시오.



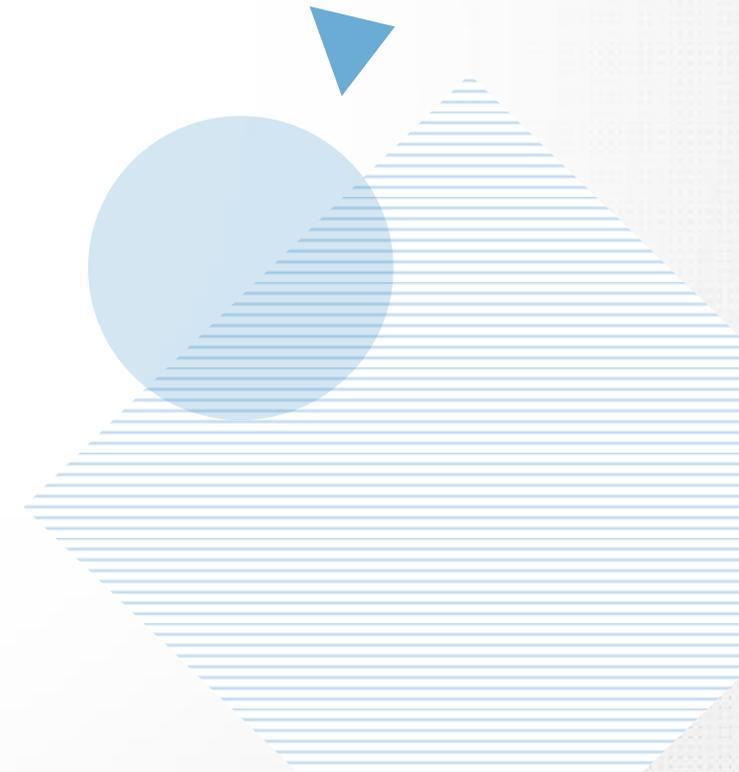
- <사원>테이블에서 주소를 기준으로 오름차순 정렬시켜서 상위 2개의 튜플만 표시하시오.
 - <사원>테이블에서 부서를 기준으로 오름차순 정렬하고 같은 부서인 경우에는 '이름'을 기준으로 내림차순 정렬시키시오.
 - <사원>테이블에서 부서별 급여의 평균을 구하시오.
 - <사원>테이블에서 부서별 튜플의 개수를 구하시오.
 - <사원>테이블에서 급여가 200 이상이고 사원이 2명 이상인 부서의 튜플의 수를 구하시오.
 - 경력 8년 초과인 사원의 이름, 부서, 특기, 경력을 검색하시오..
- 

- <사원>테이블에서 급여가 200 이상이고 사원이 2명 이상인 부서의 튜플의 수를 구하시오.
- 경력이 8년 초과인 사원의 이름, 부서, 특기, 경력을 검색하시오..



➤ JOIN의 개념

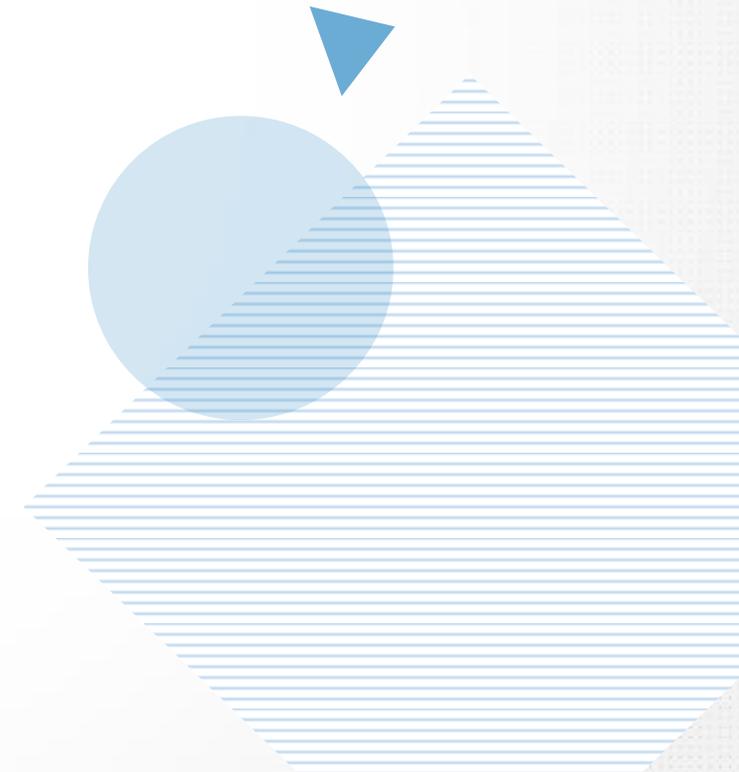
- INNER JOIN : Equi join, cross join
- OUTER JOIN : left, right, full
- SELF JOIN



1. SQL문장에서 group by 절에 의해 선택된 그룹의 탐색 조건을 지정할 수 있는 것은?

- ① Having
- ② Order by
- ③ Union
- ④ Join

정답 1



문혜영 교수

제3과목 데이터베이스 구축

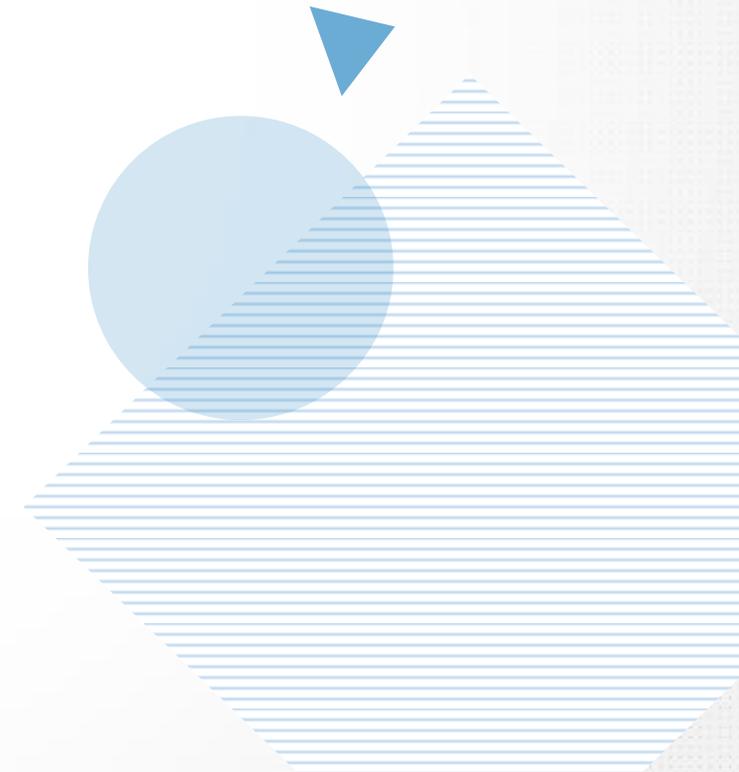
12 SQL 활용 A



프로시저(Procedure)

- 프로시저의 개요
- 프로시저 생성
- 프로시저 실행
- 프로시저 제거

```
DECLARE  
BEGIN  
  •CONTROL  
  •SQL  
  •EXCEPTION  
  •TRANSACTION  
END
```



트ริ거(Trigger)

▶ 트리거(Trigger)의 개요

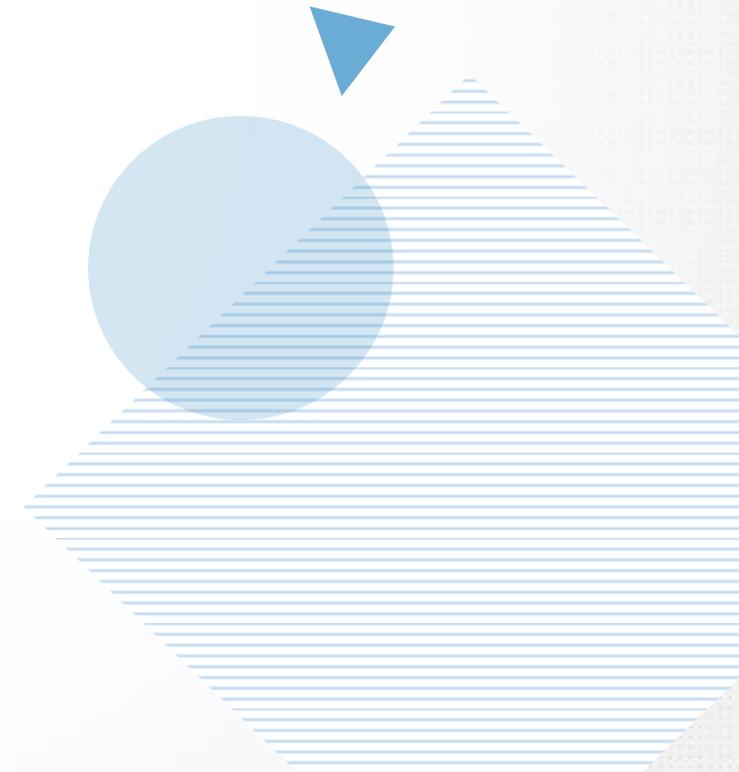
▶ 트리거의 구성

▶ 트리거의 생성

▶ 트리거의 제거

```
DECLARE  
EVENT  
BEGIN  
    •CONTROL  
    •SQL  
    •EXCEPTION  
END
```

```
CREATE TRIGGER  
  
REFERENCING AS  
FOR EACH ROW  
  
BEGIN  
  
END;
```



🔗 사용자 정의 함수

- 사용자 정의 함수의 개요
- 사용자 정의 함수의 구성
- 사용자 정의 함수 생성
- 사용자 정의 함수 실행
- 사용자 정의 함수 제거

```
DECLARE
```

```
BEGIN
```

- CONTROL
- SQL
- EXCEPTION
- RETURN

```
END
```

```
CREATE [OR REPLACE]
```

```
FUNCTION
```

```
BEGIN
```

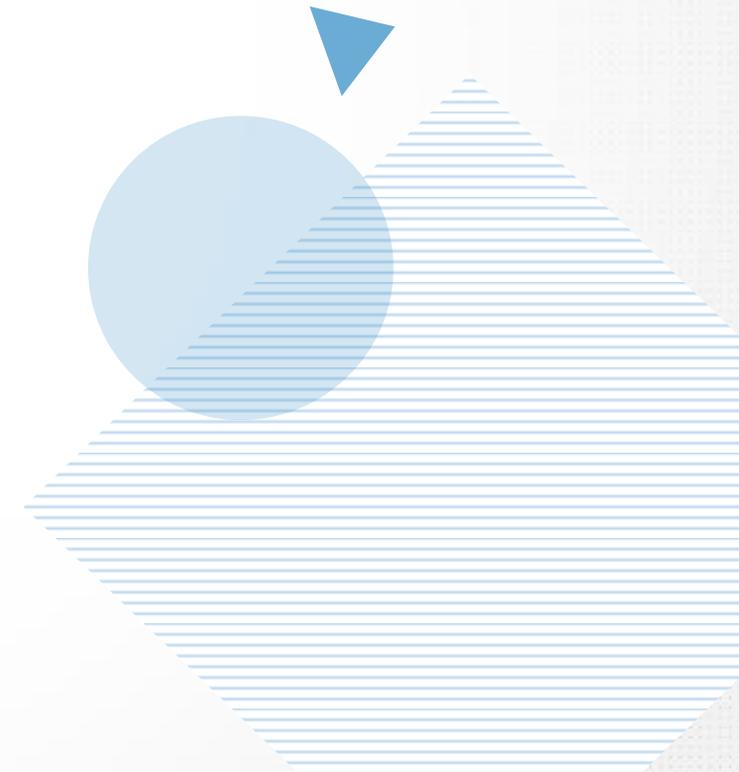
```
    사용자 정의 함수 BODY;  
    RETURN 반환값;
```

```
END;
```

1. 프로시저를 실행 또는 호출하기 위한 표기형식에 해당하지 않는 것은?

- ① EXEC [프로시저명];
- ② EXECUTE [프로시저명];
- ③ LOAD [프로시저명];
- ④ CALL [프로시저명];

정답 3



문혜영 교수

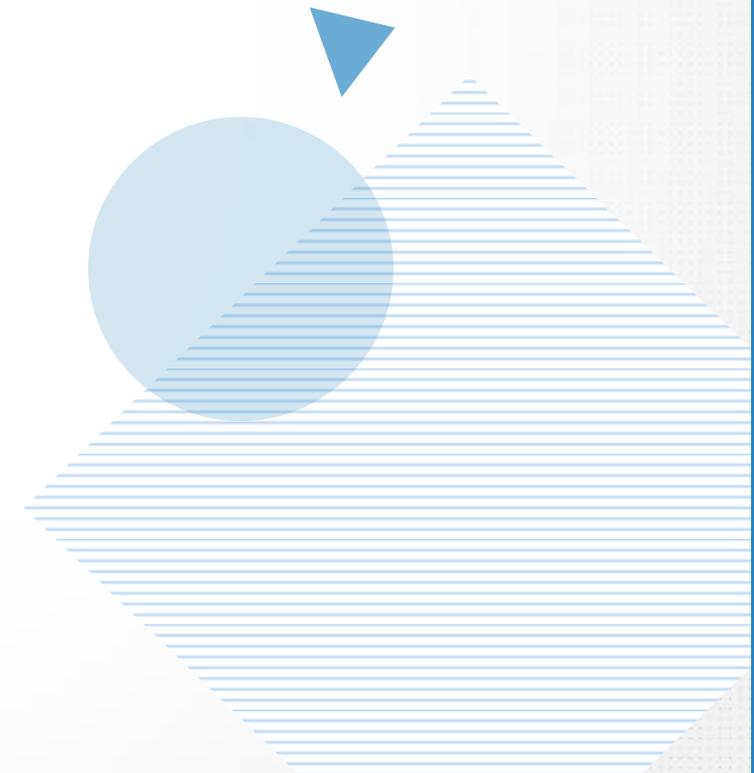
제3과목 데이터베이스 구축

13 SQL 활용 B

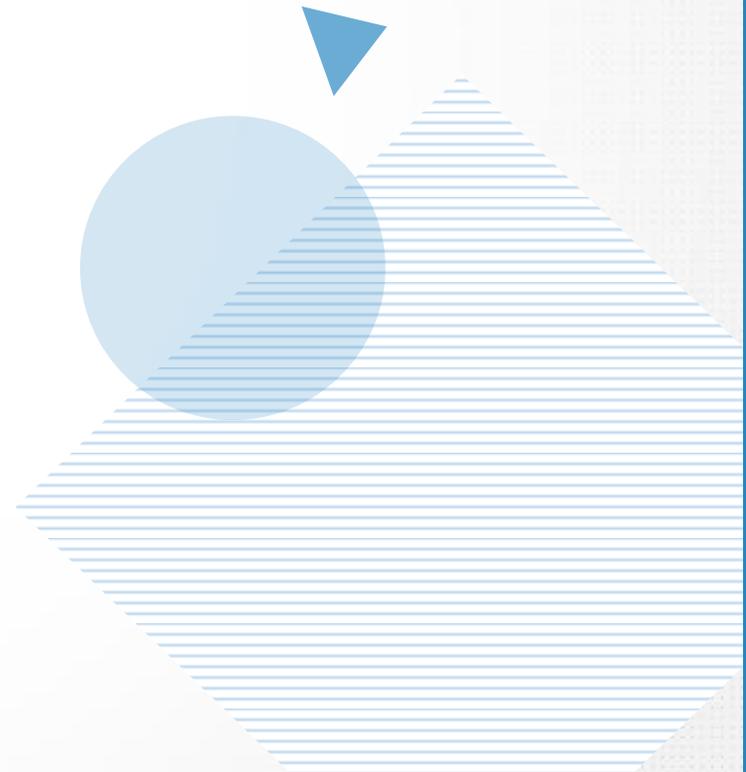


DBMS 접속 기술

- DBMS 접속의 개요
- DBMS 접속 기술
- 동적 SQL(Dynamic SQL)



- SQL 테스트의 개요
- 단문 SQL 테스트
- 절차형 SQL 테스트



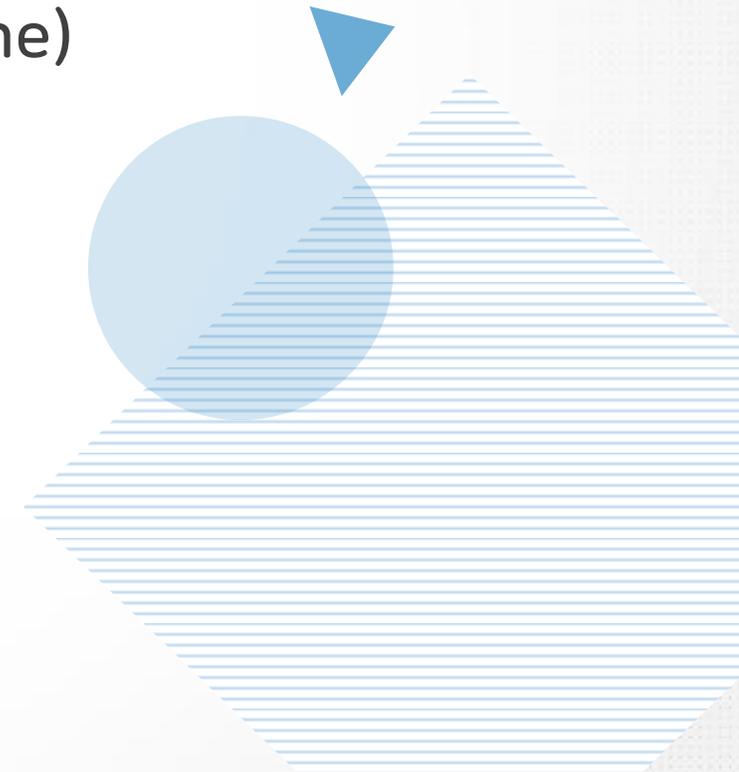
ORM(Object-Relational Mapping)

➤ ORM(Object-Relational Mapping)의 개요

➤ ORM 프레임워크

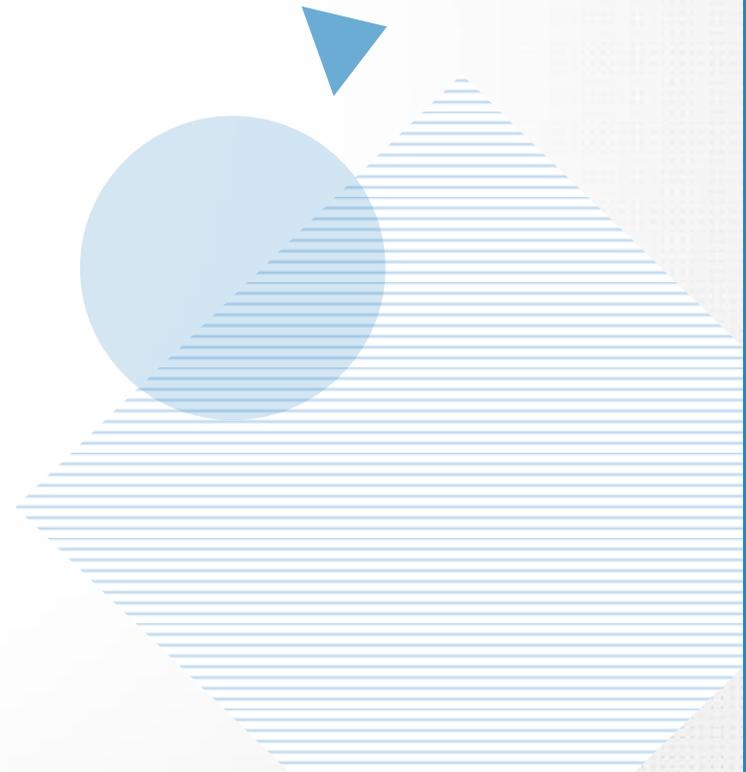
- JAVA(Hibernate, EclipseLink), C++(ODB), Python(Django), iOS(DatabaseObjects), .NET(Nhibernate), php(Doctrine)

➤ ORM의 한계



🔄 쿼리 성능 최적화

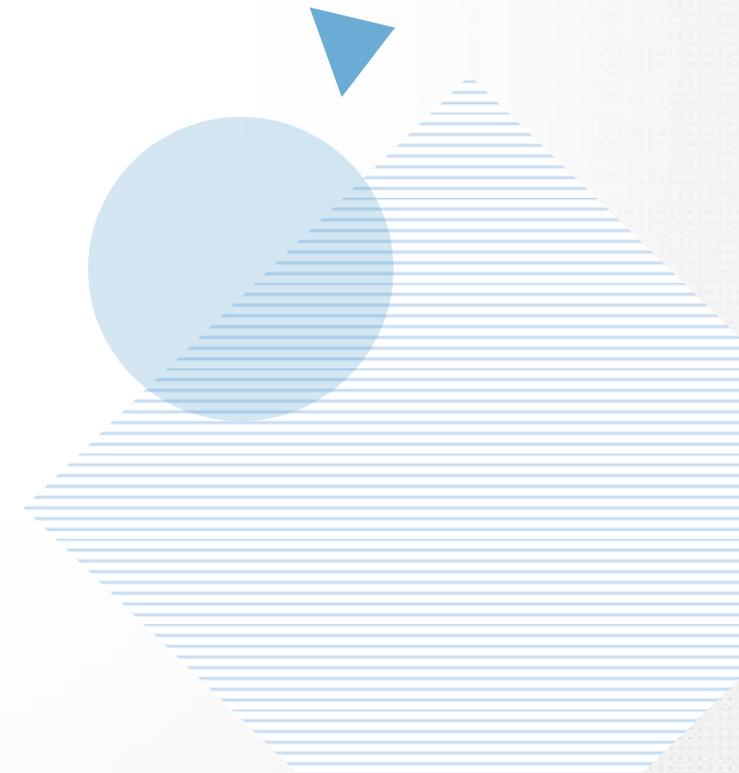
- 쿼리 성능 최적화의 개요
- 실행 계획(Execution Plan)
- 쿼리 성능 최적화



1. 객체지향의 객체 모델을 데이터베이스와 연결시키기위해 여러가지 기능들을 제공하는 소프트웨어를 무엇이라 하는가?

- ① DBMS
- ② Transaction
- ③ ORM Framework
- ④ API

정답 3



제3과목 데이터베이스 구축

14 데이터 전환 A



▶ 데이터 전환의 정의

- 운영 중인 기존 정보 시스템에 축적되어 있는 데이터를 추출하여 새로 개발한 정보 시스템에서 운영 가능하도록 변환한 후 적재하는 일련의 과정

▶ 데이터 전환 계획서

- 데이터 전환이 필요한 대상을 분석하여 데이터 전환 작업에 필요한 모든 계획을 기록하는 문서
- 주요 항목
 - 데이터 전환 개요, 데이터 전환 대상 및 범위, 데이터 전환 환경 구성, 데이터 전환 조직 및 역할, 데이터 전환 일정, 데이터 전환 방안, 데이터 정비 방안, 비상 계획, 데이터 복구 대책

🔄 데이터 전환 계획서 작성

➤ 데이터 전환 개요

- 데이터 전환 목표는 간단하고 명료하게 정의
- 데이터 전환 작업을 위한 전제 조건이나 제약 사항을 정확히 기술

➤ 데이터 전환 대상 범위

- 단위 업무별로 데이터 전환 대상 정보, 해당 업무에 사용되는 Table의 수, 데이터 크기

➤ 데이터 전환 환경 구성

- 원천 시스템 구성도
- 목적 시스템 구성도
- 전환 단계별 DISK 용량 산정



④ 데이터 전환 계획서 작성

➤ 데이터 전환 조직 및 역할 작성

- 데이터 전환을 수행하고 결과를 검증할 작업자와 작업자별 역할을 최대한 상세히 기술한다.

➤ 데이터 전환 일정 작성

- 데이터 전환 및 검증 작업별로 상세하게 일정을 수립하여 작성한다.



데이터 전환 방안

➤ 데이터 전환 방안

➤ 데이터 전환 규칙

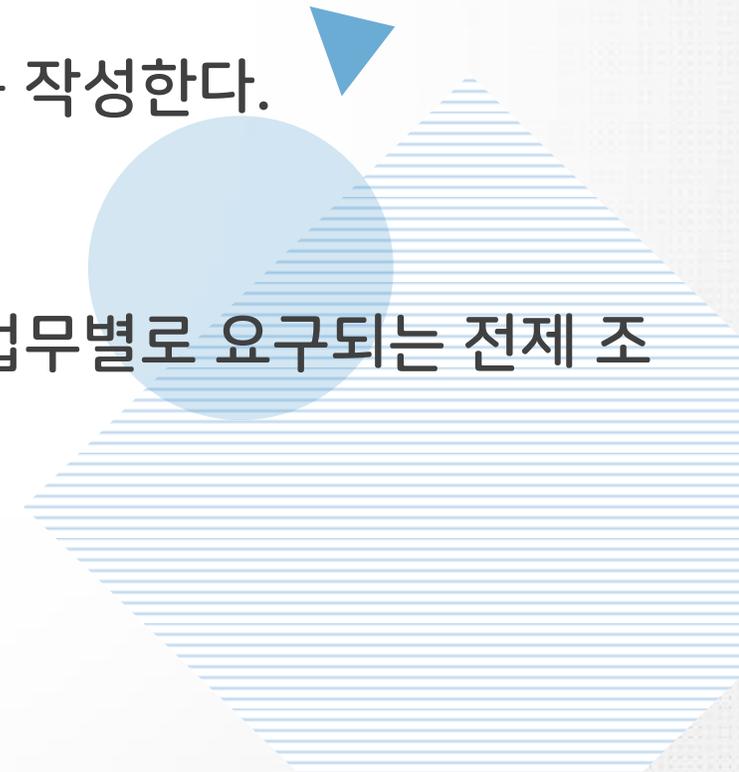
- 데이터 전환 과정에서 공통적으로 적용해야 할 규칙들을 기술한다.

➤ 데이터 전환 절차

- 데이터 전환 절차 수립 시 작업의 이해를 위해 데이터 흐름도를 작성한다.

➤ 데이터 전환 방법

- 단위 업무별로 데이터 전환 방법을 기술하되, 데이터 전환 시 업무별로 요구되는 전제 조건도 함께 기술한다.



데이터 전환 방안

▶ 데이터 전환 설계

- 업무별로 전환 대상과 전환 제외 대상을 기술한다.

▶ 전환 프로그램 개발 및 테스트 계획

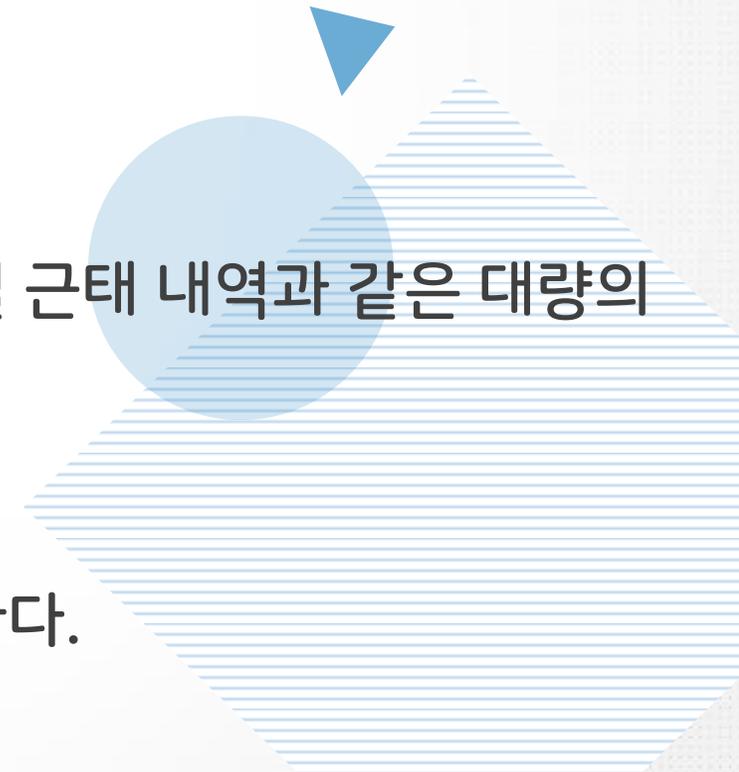
- 전환 프로그램은 목록별로 프로그램 입력 정보, 중간 생성 정보, 출력 정보, 프로그램 위치, 담당자 등을 포함하여 작성한다.

▶ 데이터 전환 계획

- 데이터 전환 시간들을 단축하기 위해 일자별 거래 내역, 일자별 근태 내역과 같은 대량의 데이터 테이블은 사전에 전환한다.

▶ 데이터 검증 방안

- 단계별 데이터 전환 검증 방안을 수립한 후 관련 내용을 기술한다.



▶ 데이터 검증

- 원천 시스템의 데이터를 목적 시스템의 데이터로 전환하는 과정이 정상적으로 수행되었는지 여부를 확인하는 과정

▶ 검증 방법에 따른 분류

- 로그 검증, 기본 항목 검증, 응용 프로그램 검증, 응용 데이터 검증, 값 검증

▪ 검증 단계에 따른 분류

- 추출, 전환, DB 적재, DB 적재 후, 전환 완료 후



오류 데이터 측정 및 정제

➤ 오류 데이터 측정 및 정제의 개요

- '데이터 품질 분석' - '오류 데이터 측정' - '오류 데이터 정제' 순으로 진행

➤ 데이터 품질 분석

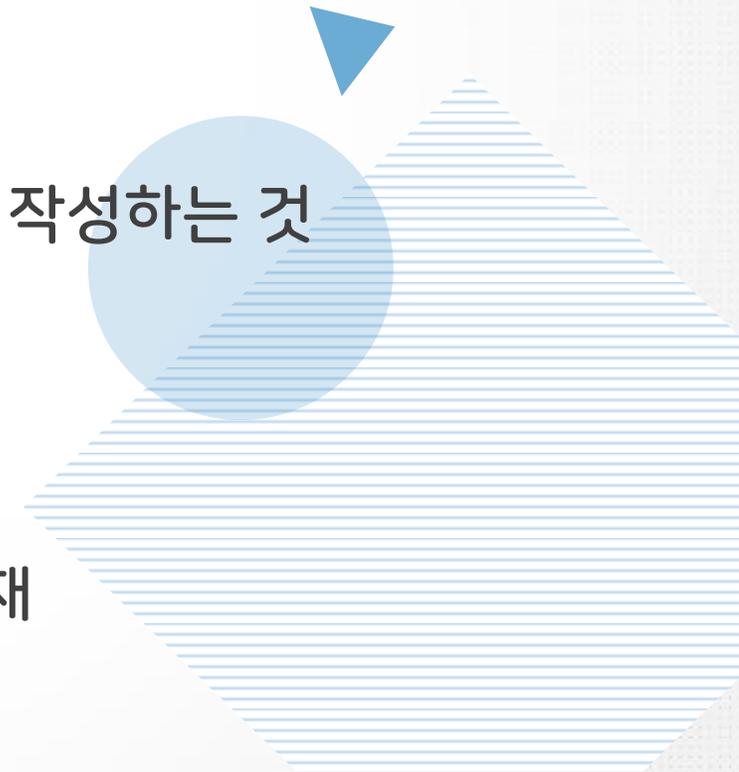
- 원천 및 목적 시스템 데이터의 정합성 여부를 확인하는 작업

➤ 오류 데이터 측정

- 정상 데이터와 오류 데이터의 수를 측정하여 오류 관리 목록을 작성하는 것

➤ 오류 데이터 정제

- 원천 데이터를 정제하거나 전환 프로그램을 수정하는 것
- 오류 데이터 분석 : 오류 상태, 심각도, 해결 방안을 확인 및 기재



데이터 정제요청서 및 정제보고서

▶ 데이터 정제요청서의 개요

- 원천 데이터의 정제와 전환 프로그램의 수정을 위해 데이터 정제와 관련된 전반적인 내용을 문서로 작성하는 것

▶ 데이터 정제 요건 목록 작성

- 정제 유형 : 완전성, 유효성, 일치성, 유일성, 기타

▶ 데이터 정제요청서 작성

- 데이터 전환시 발생한 오류의 수정을 위한 정제 요청의 전반적인 내용들을 작성

▶ 데이터 정제보고서의 개요 및 작성

- 정제 요청 데이터와 정제된 데이터 항목을 육안으로 직접 비교하여 확인한다.
- 오류 데이터 원인, 실제 데이터 정제 건수, 향후 대응 방안 등을 포함한다.

1. 다음 중 데이터 전환 검증에 사용되는 검증 방법이 아닌 것은?

- ① 로그 검증
- ② 기본 항목 검증
- ③ 정합성 검증
- ④ 값 검증

정답 3

검증 방법에 따른 분류에는 응용 프로그램 검증과 응용 데이터 검증도 있다.



1. 다음 중 오류 데이터 상태에 대한 설명으로 가장 옳지 않은 것은?

- ① Open은 오류가 보고되었지만 아직 분석되지 않은 상태를 의미한다.
- ② Assigned는 오류의 영향 분석 및 수정을 위해 개발자에게 오류를 할당한 상태를 의미한다.
- ③ Fixed는 개발자가 오류를 수정한 상태를 의미한다.
- ④ Classified는 수정이 불가능한 오류 상태를 의미한다.

정답 4

Classified는 보고된 오류를 관련자들이 확인했을 때 오류가 아니라고 확인된 상태를 의미한다.

