

뷰

• 개념

뷰 = 논리 테이블, 사용자에게 사용관점에서 테이블과 동일

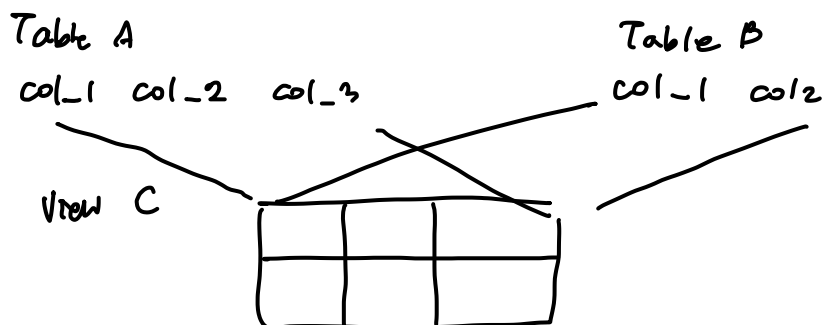


Table A, B → 물리테이블

View C → 두개의 테이블로 만든 뷰

뷰는 하나의 물리테이블로부터 생성 가능, 다수의 테이블 또는 다른 뷰 이용 가능

복와 같은 결과 위해 조인 가능 뷰 만들어져 있으면 조인 없이 하나의 테이블을 대상으로 하는 단순한 질의어 사용 가능

• 특징

- 기본테이블로부터 유도된 테이블로 기본 테이블과 같은 형태의 구조.

- 뷰는 가상테이블로 물리적인 행이 존재하지 않음

- 데이터의 논리적 독립성 제공

- 관리 용이 & 명명문이 간단

- 데이터를 안전하게 보호하는 효과적인 방법

• 목적

단순한 질의어를 사용하도록 쉽게 함 → `select * from view 이름;` 으로 가능.

~) From 절에 있는 하나의 뷰로 복수의 테이블 대체 가능, 중요한 데이터 인복만 제공

- 장 단점

• 장점

논리적 독립성 제공 : 뷰는 논리테이블 (테이블의 구조를 변경해도 뷰는 사용하는 응용 프로그램은 변경 필요 X)

사용자 데이터 관리 용이 : 복수 테이블에 존재하는 여러 종류의 데이터에 대해 단순한 질의어 사용이 가능

데이터 보안이 용이: 중요 보안 데이터는 저장 중인 테이블에서 접근 불허

해당 테이블의 일부 정보만을 볼 수 있는 뷰는 접근 허용

보안 데이터에 대한 접근 제어 가능

• 단점

뷰 자체 인덱스 증가 : 인덱스는 물리적으로 저장된 데이터로 대상으로 하기 때문에 논리적 관점인 뷰 자체에 인덱스 가지지 X.

뷰 정의 변경 불가 : 뷰의 정의를 변경하려면 뷰를 삭제하고 재생성

데이터 변경 제약 존재: 뷰의 내용에 대한 삽입, 삭제, 변경에 제약이 있음

• 뷰 생성

CREATE VIEW 뷰 이름 권한 목록 **AS** 데이터 조회 쿼리;

→ select 쿼리

* select 쿼리에 union, order by 불가, 컬럼명 지정 X → 2 테이블 컬럼 2개

• 뷰 삭제 (변경)

뷰 정의 변경 X, 뷰 이름, 쿼리를 변경 Y → 뷰 삭제와 재생성은 둘래 가능

Drop VIEW 뷰 이름;

뷰를 통해 쿼리 가능한 데이터는 변경 가능

기본키 (PK)에 해당하는 컬럼이 뷰에 정의되어 있지 않은 경우는 insert X

1. 2 ^토 3 ^수 4 ^목 5 ^금 6 ^토 7 ^일 8 ^월 9 ^화 10 ^수 11 ^목 12 ^금 13 ^토 14 ^일 15 ^월 16 ^화 17 ^수 18 ^목 19 ^금 20 ^토 21 ^일 22 ^월 23 ^화 24 ^수 25 ^목 26 ^금 27 ^토 28 ^일 29 ^월 30 ^화 31 ^수

MySQL, Oracle, PostgreSQL, MS SQL Server

MySQL / Rank / Row-Number / Dense Rank / Coalesce

Union all / Minus / Union

조인 방식 → (42)

Connect by → (35)

데이터 타입

문자열 → (24)

숫자형 데이터 → (30) 날짜형 데이터 → (31)

DDL → 34 DCL → 35

MySQL 쿼리 → 39

MySQL DB → 41

ROUND / ABS / FLOOR / TRUNC / CEIL ... (44)