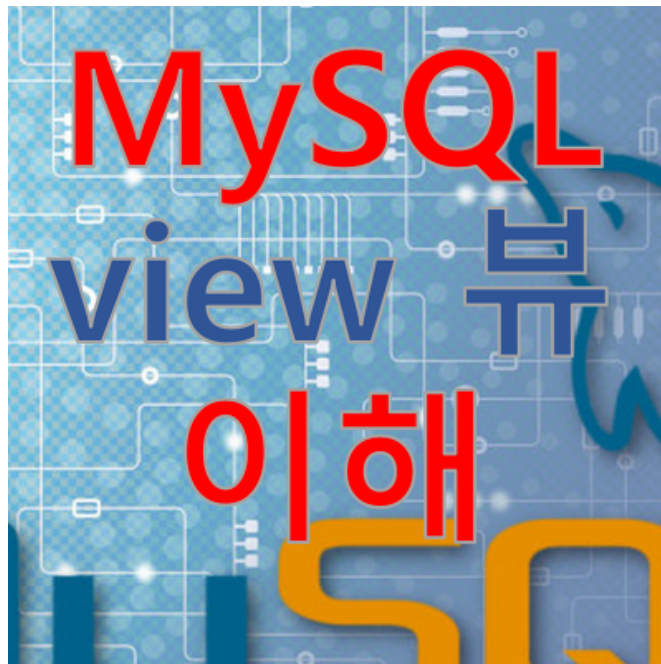


view 뷰에 대한 이해 SQL 독학 강의#25편

sTricky 2020. 7. 27. 14:36

view 뷰에 대한 이해 SQL 독학 강의#25편



[콘텐츠 index](#)

0. view란 무엇인가?

1. view를 사용하는 이유

2. view의 특징

3. view 생성 및 삭제 문 예제

4. view 실행 원리 및 예제

전편 강의 보러 가기

##

<https://stricky.tistory.com/310>

이번 SQL 독학 강의의 주제는 view 입니다.

많이 들어본 view, 뷰란 도대체 무엇일까요?

뷰에 대한 이해를 높이고 어떻게 사용을 하고 어떤 의미로 받아들여야 하는지에 관해서 설명을 드리겠습니다.

0. view란 무엇인가?

뷰는 가상의 테이블이라고 의미하면 됩니다.

이미 우리가 지난 시간에 select 서브쿼리에 대해서 공부할때 inline view (인라인 뷰) 라는것에 관해서 공부를 했었습니다.

```

SELECT  cs.name, SUM(od.saleprice) "total"
FROM    (SELECT  custid, name
        FROM    Customer
        WHERE   custid <= 2) cs,
        Orders od
WHERE   cs.custid = od.custid
GROUP BY cs.name ;
  
```

인라인 뷰

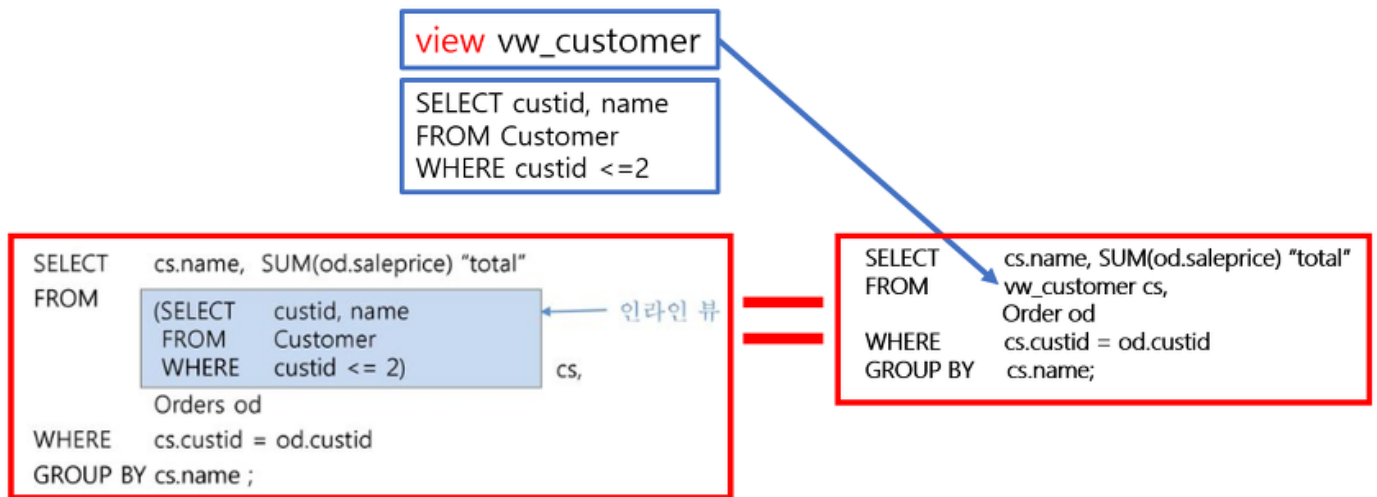
from 절에 들어가는 (,) (괄호) 안에 있는 select문을 다시 한번 상기 시켜 보세요.

이것을 inline view 라고 이야기하는데, 말그대로 inline, SQL문 안에 있는 view 라는 뜻 입니다.

맞습니다. view(뷰)를 위에서 가상의 테이블이라고 했는데, 말그대로 데이터는 없고, SQL만 저장되어있는 object를 말합니다.

view를 select 하게 되면 view가 가지고 있는 SQL문이 실행이 되는것과 같습니다.

바로 아래 그림과 마찬가지로 말이죠!



1. view를 사용하는 이유

그럼 그냥 테이블을 쓰던지 inline view로 쓰면 되지 왜 굳이 view라는 object를 만들어서 따로 관리를 하는것 일까요?

이유는 보안과 함께 사용자의 편의성을 높이기 위해서 입니다.

또한 여러테이블을 조인 할 시에 view 뷰를 통하면 SQL을 어느정도 간소화 시킬 수 있습니다. 또한 복잡한 SQL을 편리하게 재 생성 할 수 있는 장점이 있습니다.

예를 들어 여러개의 schema나 user로 구성된 database가 있는데, 특정 유저가 A라는 스키마의 테이블을 보는 권한이 없고, 더 불어 보안상 봐서도 안된다라고 가정 한 상태에서, 그 유저가 A 스키마내 특정 테이블의 어떤 정보는 업무상 꼭 필요하다 할때, 이럴때 view를 생성하여 꼭 필요한것만 보여주고, 전체 정보는 공개하지 않게 할 수 있겠죠.

이는 지난시간에 함께 공부했던 synonym의 필요성과도 일맥상통한데, synonym은 특정 테이블의 전체를 보여주되 일부는 보이지 못하게 못한다면, view는 전체 또는 일부만 공개 할 수 있는 장점이 있으며, join이라던지 다른 SQL 작성 기법을 사용하여 여러 테이블의 데이터를 함께 보여줄 수 있는 장점도 있습니다.

어떻게 생각해보면 synonym보다 훨씬 편리하게 사용 할 수 있는 object 입니다.



2. view의 특징

view 뷰는 기본적으로 일반 테이블과 같은 형태를 가지고 있고, SQL문으로 조작하는 방법 또한 거의 흡사 합니다.

또한 뷰는 데이터의 논리적 독립성을 제공하며, 필요한 데이터만 뷰로 정의하여 사용하기 때문에 관리가 용이하고, SQL 명령문이 간단해 집니다.

만들어진 뷰의 기본 테이블의 기본키(속성)를 포함하여 뷰를 생성하면 삽입, 삭제, 갱신, 연산이 가능 합니다. 또한 한번 정의가 된 뷰의 경우 다른 뷰의 기본 데이터가 될 수 있으며, 뷰에 정의되어 있는 기본 테이블이나 뷰를 삭제 하게 되면 해당 데이터를 기초로한 다른 뷰들이 자동으로 삭제가 됩니다.

뷰에서는 ALTER 명령어를 사용 할 수 없습니다. 뷰의 내용을 수정 하고 싶으면 drop & create를 반복 하여야 합니다.

뷰는 원본 테이블과 같은 이름으로 생성 할 수 없습니다. 실무에서는 대체적으로 "vw_" 등의 접미사나 접두사를 붙여 해당 데이터셋이 뷰라는것을 명시 해주고 있습니다.

3. view 생성 및 삭제 문 예제

```
-- 생성문

CREATE VIEW 뷰이름 AS SELECT 구문;

-- 삭제문

DROP VIEW 뷰이름;
```

4. view 실행 원리 및 예제

1) 아래와 같은 view의 생성문이 있다고 가정 합니다.

```
SELECT name,

        money_received,

        money_sent,

        (money_received - money_sent) AS balance,

        address,

        ...

FROM table_customers c
```

```
JOIN accounts_table a  
  
ON a.customer_id = c.customer_id
```

2) 위 view를 이용하여 간단한 SQL을 실행합니다.

```
SELECT name,  
  
       balance  
  
FROM accounts_view;
```

3) DBMS 옵티마이저가 인식하는 SQL은 아래와 같습니다.

```
SELECT name,  
  
       balance  
  
FROM (SELECT name,  
  
            money_received,  
  
            money_sent,  
  
            (money_received - money_sent) AS balance,  
  
            address,  
  
            ...  
  
FROM table_customers c JOIN accounts_table a  
  
ON a.customer_id = c.customer_id );
```

view 뷰에 관해서 간단하게 알아보았습니다.

생각보다 쉽지 않으신가요?

