

# 11장 뷰

이번 장에서는 뷰의 개념과 함께 뷰를 생성하는 방법을 살펴보고, 생성된 뷰가 내부적으로 어떻게 처리되는지 살펴보겠습니다.

## 학습 내용

- ❖ 뷰의 개념과 뷰 사용하기
- ❖ 다양한 뷰 옵션

## 학습목표

- ❖ 뷰의 개념과 내부 구조를 이해할 수 있습니다.
- ❖ 뷰를 사용하는 이유를 알 수 있습니다.
- ❖ 보안과 복잡한 쿼리문을 대신할 뷰를 생성할 수 있습니다.
- ❖ 뷰를 제거 및 변경할 수 있습니다.

## 01. 뷰의 개념과 사용하기

- ❖ 뷰(View)는 한마디로 물리적인 테이블을 근거한 논리적인 가상 테이블이라고 정의할 수 있습니다.
- ❖ 디스크 저장 공간이 할당되지 않습니다.
- ❖ 사용 방법은 테이블에서 파생된 객체 테이블과 유사합니다.

## 뷰 생성

### ❖ 뷰를 생성하기 위한 형식

```
CREATE [OR REPLACE] [FORCE | NOFORCE] VIEW view_name  
[(alias, alias, alias, ...)]  
AS subquery  
[WITH CHECK OPTION]  
[WITH READ ONLY];
```

## 뷰의 종류

- ❖ 뷰는 뷰를 정의하기 위해서 사용되는 기본 테이블의 수에 따라 단순 뷰(Simple View)와 복합 뷰(Complex View)로 나뉩니다.
- ❖ 단순 뷰(simple view)는 하나의 기본 테이블로 생성한 뷰이며 DML 명령문을 실행할 수 있습니다. DML 명령문의 처리 결과는 기본 테이블에 반영됩니다.
- ❖ 복합 뷰(complex view)는 두개 이상의 기본 테이블로 생성한 뷰입니다.

## 뷰의 필요성

- ❖ 뷰를 사용하는 이유는 보안과 사용의 편의성 때문
- ❖ 보안을 위해 뷰 활용하기

```
create view v_emp_sample  
as  
select eno, ename, job, manager, dno  
from emp_second;
```

- ❖ 정보를 손쉽게 얻기 위해 뷰 활용하기

```
create view v_emp_complex2  
as  
select e.eno, e.ename, e.salary, dno, d.dname, d.loc  
from emp_second e natural join dept_second d;
```

## 뷰의 처리 과정

### ❖ USER\_VIEWS 데이터 사전 살피기

```
SELECT VIEW_NAME, TEXT  
FROM USER_VIEWS;
```

- ❖ 뷰는 데이터를 저장하고 있지 않은 가상 테이블이므로 실체가 없습니다. 뷰가 테이블처럼 사용될 수 있는 이유는 뷰를 정의할 때 CREATE VIEW 명령어 다음의 AS 절에 기술한 쿼리 문장 자체를 저장하고 있다가 이를 실행하기 때문입니다.



## 뷰의 처리 과정

- ❖ **USER\_VIEW 데이터 사전에서 뷰에 대한 정의를 조회합니다.**
- ❖ **기본 테이블에 대한 뷰의 접근 권한을 살핍니다.**
- ❖ **뷰에 대한 질의를 기본 테이블에 대한 질의로 변환합니다.**
- ❖ **기본 테이블에 대한 질의를 통해 데이터를 검색합니다.**
- ❖ **검색된 결과를 출력합니다.**

## 뷰의 처리 과정

- ❖ 뷰에 대해서 질의를 하면 오라클 서버는 USER\_VIEWS에서 뷰를 찾아 이를 정의할 때 기술한 서브 쿼리문을 뷰 위치로 가져옵니다.

```
SELECT *  
FROM v_emp_job;
```



```
select eno, ename, dno, job  
from emp_second  
where job like 'SALESMAN';
```

## 다양한 뷰

### ❖ 뷰에 그룹함수 사용하기(별칭 사용)

```
create view v_emp_salary  
as  
select dno, sum(salary) as "sal_sum", avg(salary) as "sal_avg"  
from emp_second  
group by dno;
```

## 뷰 제거하기

- ❖ 뷰가 더 이상 필요 없을 때는 DROP VIEW 문을 이용하여 제거

```
DROP VIEW {view} [,..n]
```

- ❖ 뷰는 실체가 없는 가상 테이블이기 때문에 뷰를 삭제한다는 것은 USER\_VIEWS 데이터 디렉터리에 저장되어 있는 뷰의 정의를 삭제하는 것을 의미합니다.
- ❖ 따라서 뷰를 삭제해도 뷰를 정의한 기본 테이블의 구조나 데이터에는 전혀 영향을 주지 않습니다.

```
drop view v_emp_job;
```

## 다양한 뷰 옵션

```
CREATE [OR REPLACE] [FORCE | NOFORCE] VIEW view_name  
[(alias, alias, alias, ...)]  
AS subquery  
[WITH CHECK OPTION]  
[WITH READ ONLY];
```

### ❖ CREATE OR RELPACE VIEW

- 이미 존재하는 뷰에 대해서 그 내용을 새롭게 변경하여 재 생성

### ❖ FORCE

- 기본 테이블의 존재 여부에 상관없이 뷰 생성

## 다양한 뷰 옵션

### ❖ WITH CHECK OPTION

- WITH CHECK OPTION을 사용하면, 해당 뷰를 통해서 볼 수 있는 범위 내에서만 UPDATE 또는 INSERT가 가능합니다.

### ❖ WITH READ ONLY

- WITH READ ONLY를 사용하면 해당 뷰를 통해서는 SELECT만 가능하며 INSERT/UPDATE/DELETE를 할 수 없게 됩니다.
- 만약 이것을 생략한다면, 뷰를 사용하여 추가, 수정, 삭제(INSERT/UPDATE/DELETE)가 모두 가능합니다.