

221004 수업 -1

| | |
|--------|--------------------------|
| 🕒 작성일시 | @2022년 10월 4일 오전 8:53 |
| ≡ 키워드 | sequence |
| ▼ 유형 | |
| 🔗 자료 | |
| ☑ 복습 | <input type="checkbox"/> |

Sequence

- 자동으로 번호를 생성하는 객체
- 중복값이 없고 순차적인 값을 제공하는 객체
- 공유 객체
 - 테이블에 종속되지 않음
 - 하나의 시퀀스는 여러 곳에서 자유롭게 사용가능

문법

```
CREATE SEQUENCE sequence_name  
[INCREMENT BY n]  
[START WITH n]  
[{MAXVALUE n | NOMAXVALUE}]  
[{MINVALUE n | NOMINVALUE}]  
[{CYCLE | NOCYCLE}]  
[{CACHE | NOCACHE}];
```

```
create sequence board_num;  
  
select board_num.nextval from dual;
```

- NEXTVAL 는 다음 사용 가능한 SEQUENCE 값을 반환 한다.

```
select board_num.currval from dual;
```

- CURRVAL 은 현재 SEQUENCE 값을 얻는다.
- 현재까지 채번한 번호 확인(마지막)

Tip

MS-SQL

```
create table board(boardnum int, identity(1,1) ... title
insert into(title) values('방가');
```

-MS-SQL : 2012 버전 (sequence)

MySQL

```
create table board(boardnum int auto_increment, ... title)
insert into board(title) value('방가');
```

옵션

```
create sequence seq_num
start with 10
increment by 2;
```

- 10부터 시작해서 2씩 증가

rownum 의사컬럼

- 실제 물리적으로 존재하는 컬럼이 아니고 논리적으로 존재
- 실제로 테이블에 컬럼으로 존재하지 않지만 내부적으로 행 번호를 부여하는 컬럼

```
select rownum as 순번, empno, ename from emp;
```

- select 한 결과에 순번을 부여 ...

Top-n 쿼리

- 정렬된 기준으로 위에서 몇개 추출
- 테이블에서 조건에 맞는 상위(Top)에서 레코드(row) n개 추출
- 오라클에는 없음
 - rownum : 순번을 부여 해 특정 조건에 맞게 top 쿼리 실행 가능
 1. 정렬의 기준을 정의(선행)
 2. 정렬된 데이터를 기준으로 rownum 설정하고 조건을 걸고 데이터 추출

```
--1단계 : 정렬의 기준 정의
select *
from (
    select *
    from emp
    order by sal desc
) e;
```

```
--2단계 : 정렬된 데이터 앞에 컬럼이 붙음
select rownum as num, e.*
from (
    select *
    from emp
    order by sal desc
) e;
```

```
-- 3단계 : 급여를 많이 받는 사원 5명
select *
from (
    select rownum as num, e.*
    from (
        select *
        from emp
        order by sal desc
    ) e
) n where num <= 5;
```

게시판 페이징 원리

```
--1단계 (기준 데이터 만들기(정렬))
--사번이 낮은 순으로 정렬

select * from employees order by employee_id asc;
```

```
--2단계 (기준 데이터에 순번 부여)
select rownum as num, e.*
from (
    select * from employees order by employee_id asc
) e
where rownum <= 50; --정렬된 데이터에 내부적으로 생성된 rownum
```

```
--3단계
select *
from (
    select rownum as num, e.*
    from (
        select * from employees order by employee_id asc
    ) e
    where rownum <= 50
) n where num >= 41;
```

WITH절

- 오라클9 이후 버전부터 사용이 가능하며 이름이 부여된 서브쿼리

```
WITH EXAMPLE AS
(
    SELECT 'WITH절' AS STR1
    FROM DUAL
)

SELECT * FROM EXAMPLE;
```

이름만으로 동명이인 찾기 → over 절

```
WITH emp_t(ename, birthdate, job) AS (  
  SELECT '스미스', '1982-01-23', 'SALESMAN' FROM dual UNION ALL  
  SELECT '스미스', '1987-04-19', 'CLERK'     FROM dual UNION ALL  
  SELECT '제임스', '1981-12-03', 'SALESMAN' FROM dual UNION ALL  
  SELECT '제임스', '1981-12-03', 'CLERK'     FROM dual UNION ALL  
  SELECT '클라크', '1981-06-09', 'MANAGER'  FROM dual  
)  
  
SELECT ename  
      , birthdate  
      , job  
      , COUNT(*) OVER(PARTITION BY ename) AS sname_cnt  
FROM emp_t
```