07. Sequence

1. Sequence

❏ 정의

　(고유한) 일련번호 생성기

❏ 용도

　의미가 없지만 고유한 일련번호에 사용
　ex) 접수번호,배송번호,등록번호

❏ [참고] 식별자 4가지 특성

　　　　① 최소성
　　　　② 고유성
　　　　③ 존재성
　　　　④ 불변성

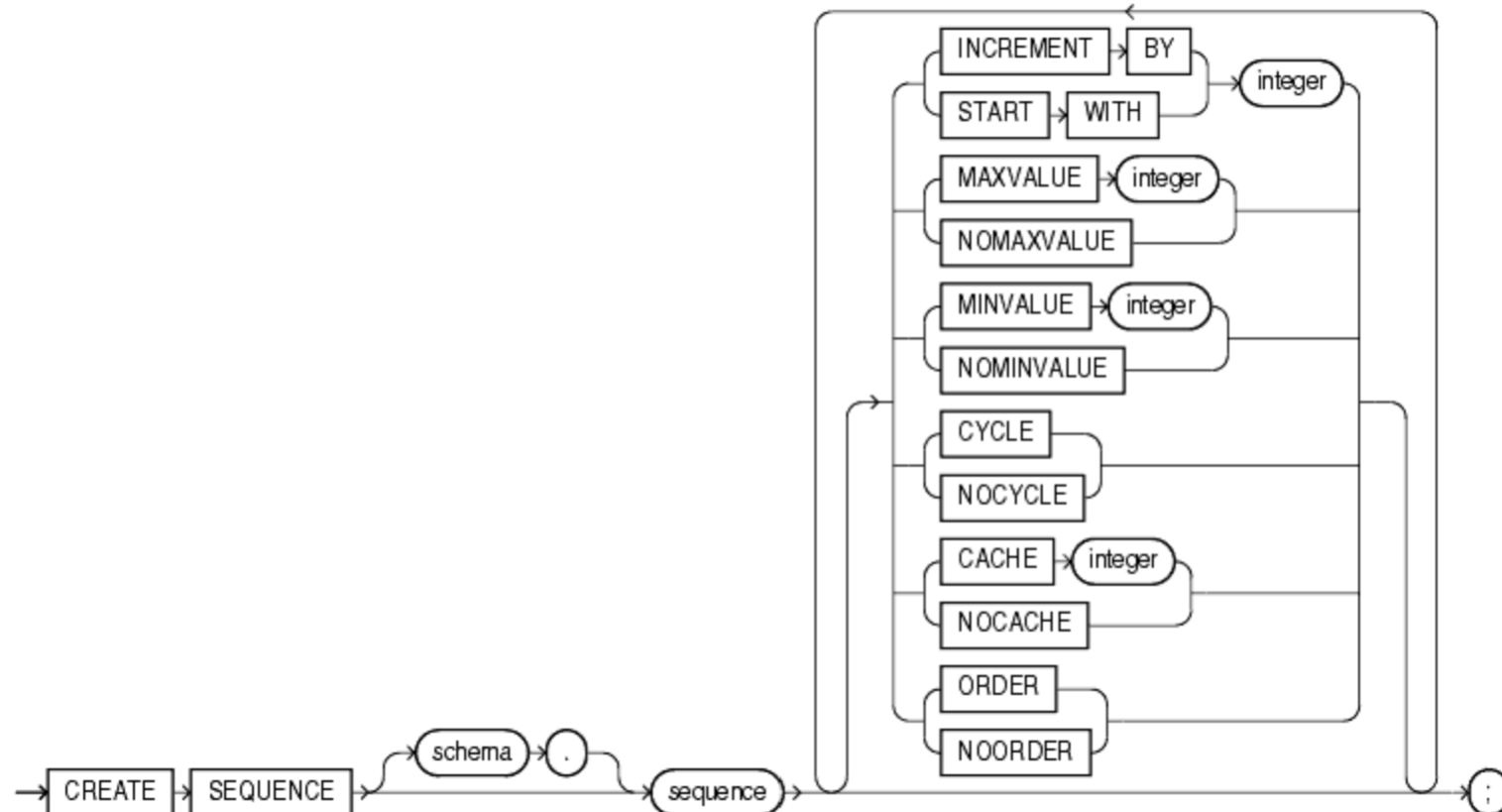❏일련번호 자동생성방식

　　　① MAX(SEQ) + 1 방식
　　　② 채번 TABLE 방식
　　　③ SEQUENCE

1. Sequence

❏ Syntax

create_sequence::=

● 1. Sequence

❏ create sequence

① CREATE SEQUENCE SCOTT.ORDER_SEQ
```
        INCREMENT BY          1
        START       WITH      1
        MAXVALUE              999999999999
        MINVALUE             1
        NOCYCLE
        CACHE                30;
```

② SELECT ORDER_SEQ.CURRVAL FROM DUAL;   // error 발생 이유는?

③ SELECT ORDER_SEQ.NEXTVAL FROM DUAL;
  SELECT ORDER_SEQ.CURRVAL FROM DUAL;   // error 발생하지 않는 이유는?
  SELECT ORDER_SEQ.CURRVAL FROM DUAL;

④ SELECT ORDER_SEQ.NEXTVAL FROM DUAL;
  SELECT ORDER_SEQ.NEXTVAL FROM DUAL;
  ROLLBACK;   // 증가된 sequence는 rollback 되는가?
  SELECT ORDER_SEQ.NEXTVAL FROM DUAL;

⑤ SELECT EMPNO,ENAME,ORDER_SEQ.NEXTVAL FROM EMP;   // Sequence 용도?

1. Sequence

❏ Session에서 sequence 테스트

Session1	Session2
⑥ SELECT ORDER_SEQ.NEXTVAL FROM DUAL;	
	⑦ SELECT ORDER_SEQ.NEXTVAL FROM DUAL;
⑧ SELECT ORDER_SEQ.NEXTVAL FROM DUAL;	
	⑨ SELECT ORDER_SEQ.NEXTVAL FROM DUAL;
⑩ SELECT ORDER_SEQ.CURRVAL FROM DUAL;	
	⑪ SELECT ORDER_SEQ.CURRVAL FROM DUAL;
⑬ SELECT ORDER_SEQ.CURRVAL FROM DUAL;	
	⑫ SELECT ORDER_SEQ.CURRVAL FROM DUAL;

// 각 Session에서 sequence object를 공유하는가?
// 각 Session에서 sequence.currval를 공유하는가?

//여러 세션(여러 Application)에서 독립적으로 동시에 사용할수 있는 공용 오브젝트

● 1. Sequence

❏ sequence 정보 조회

① CREATE SEQUENCE ORDER_SEQ2;

② SELECT * FROM USER_SEQUENCES;
  SELECT sequence_name,min_value,max_value,increment_by,cycle_flag,cache_size,last_number
  FROM USER_SEQUENCES;

  * default value
  * MAXVALUE : 28 or fewer digits

1. Sequence

❑ sequence 실습

C:\SQL\ORDER_TBL.SQL

```
DROP TABLE   SCOTT.ORDERS;


CREATE TABLE   SCOTT.ORDERS        -- ORDER 테이블명을 사용할수 없는 이유는?
(  ORDER_ID          NUMBER(12)      CONSTRAINT ORDER_ID_PK PRIMARY KEY,
   ORDER_DATE      DATE             DEFAULT SYSDATE CONSTRAINT ORDER_DATE_NN NOT NULL,
   ORDER_MODE      VARCHAR2(8)     CONSTRAINT ORDER_MODE_CHK CHECK(ORDER_MODE IN ('direct','online')),
   CUSTOMER_ID    NUMBER(6)        CONSTRAINT ORDER_CUSTOMER_ID_NN NOT NULL,
   ORDER_STATUS  NUMBER(2),        /*  0: 주문중 , 1: 주문완료, 2: 취소 , 3: 발송 4: 수취  */
   SALES_ID          NUMBER(6),        -- 판매사원번호
   CONSTRAINT ORDER_SALES_ID_FK FOREIGN KEY(SALES_ID) REFERENCES SCOTT.EMP(EMPNO)
 );


DESC SCOTT.ORDERS
```

@ C:\SQL\ORDER_TBL.SQL   // start C:\SQL\ORDER_TBL.SQL

● 1. Sequence

❏ sequence 실습

① INSERT INTO ORDERS(ORDER_ID,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,SALES_ID)
   VALUES(**ORDER_SEQ.NEXTVAL**,'direct',166,1,7499);

② INSERT INTO SCOTT.ORDERS(ORDER_ID,ORDER_DATE,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,SALES_ID)
   VALUES(**SCOTT.**ORDER_SEQ.NEXTVAL,**SYSDATE**,'online',200,3,7521);

③ COMMIT;

④ INSERT INTO ORDERS(ORDER_ID,ORDER_DATE,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,SALES_ID)
   VALUES(ORDER_SEQ.NEXTVAL,SYSDATE,'online',135,2,7844);

⑤ **ROLLBACK;**   // ORDER_SEQ.NEXTVAL도 취소 되는가?

⑥ SELECT ORDER_ID FROM ORDERS;
   SELECT ORDER_SEQ.CURRVAL FROM DUAL;

// Hole 발생
⑦ INSERT INTO ORDERS(ORDER_ID,ORDER_DATE,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,SALES_ID)
   VALUES(ORDER_SEQ.NEXTVAL,SYSDATE,'direct',135,4,7844);

⑧ // MAX+1 방식 , 문제점은 ?
   INSERT INTO scott.ORDERS(ORDER_ID,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,SALES_ID)
   VALUES(**(SELECT MAX(ORDER_ID)+1 FROM scott.ORDERS)**,'direct',335,1,7654);

● 1. Sequence

❏ 채번 테이블 방식

C:₩SQL₩ORDER_ID_TBL.SQL

```
DROP TABLE   ORDER_ID;


CREATE TABLE   ORDER_ID
(  ORDER_SEQ    NUMBER(12) );


INSERT INTO SCOTT.ORDER_ID VALUES(100);
COMMIT;
```

@ C:₩SQL₩ORDER_ID_TBL.SQL   // start C:₩SQL₩ORDER_ID_TBL.SQL

```
DECLARE
          V_ORDER_SEQ                         NUMBER(12);
BEGIN
  SELECT  ORDER_SEQ  INTO V_ORDER_SEQ  FROM  ORDER_ID FOR UPDATE;

  INSERT INTO  ORDERS(ORDER_ID,ORDER_MODE,CUSTOMER_ID,ORDER_STATUS,SALES_ID)
  VALUES(V_ORDER_SEQ,'direct',3320,2,7788);

  UPDATE ORDER_ID SET ORDER_SEQ = ORDER_SEQ +1;
  COMMIT;
END;
/
```

1. Sequence

과제

1) SEQUENCE의 START 값을 10에서 다시 시작하는 방법은?

2) 서로 다른 SESSION에서 SEQUENCE를 공유하여 사용해보고 결과 발표

3) SEQUENCE를 생성하여 00000001, 00000002와 같은 8자리 접수번호를 사용하는 SQL작성 하십시요

4) 위의 주문(ORDERS) 테이블에 추가해야할 컬럼을 고려해서 스크립트를 수정 하십시요