

데이터베이스설계 Q6

문제 1) 전자제품 번호가 가질 수 있는 값은 최소값 1, 최댓값 1000 이다. 1000부터 시작해서 10 씩 감소하는 'Q7_EL' 테이블을 생성하고, 데이터를 삽입하시오.
(단, 전자제품 번호는 sequence를 이용하여 생성한다.)



```
create table Q7_EL(
  ELECT_ID number(10),
  ELECT_NAME varchar2(30),
  ELECT_COST number(15)
);

create sequence Q7_1
increment by -10
start with 1000
maxvalue 1000
minvalue 1
cycle;

insert into Q7_EL
values(Q7_1.nextval,'iphone',1000000);

insert into Q7_EL
values(Q7_1.nextval,'mac',3500000);

insert into Q7_EL
values(Q7_1.nextval,'ipad',900000);

select * from Q7_EL;
```

스크립트 출력 x | 질의 결과 x

SQL | 인출된 모든 행: 3(0.007초)

	ELECT_ID	ELECT_NAME	ELECT_COST
1	990	iphone	1000000
2	980	mac	3500000
3	970	ipad	900000

문제 2)

- ① 위에서 생성된 “Q7_EL” 테이블에 대해 전자제품 번호 값을 30씩 감소하는 테이블로 변경하고, 데이터를 삽입하시오. (단, 전자제품 번호는 sequence를 이용하여 생성한다.)
- ② 데이터 생성에 사용한 sequence를 삭제하시오.

The screenshot shows a SQL IDE window with a script editor and a results pane. The script editor contains the following SQL code:

```
alter sequence Q7_1
increment by -30;

insert into Q7_EL
values(Q7_1.nextval, 'samsung phone', 1200000);

insert into Q7_EL
values(Q7_1.nextval, 'LG phone', 1100000);

select * from Q7_EL;
```

The results pane shows the output of the query, displaying 5 rows of data:

ELECT_ID	ELECT_NAME	ELECT_COST
1	990 iphone	1000000
2	980 mac	3500000
3	970 ipad	900000
4	940 samsung phone	1200000
5	910 LG phone	1100000

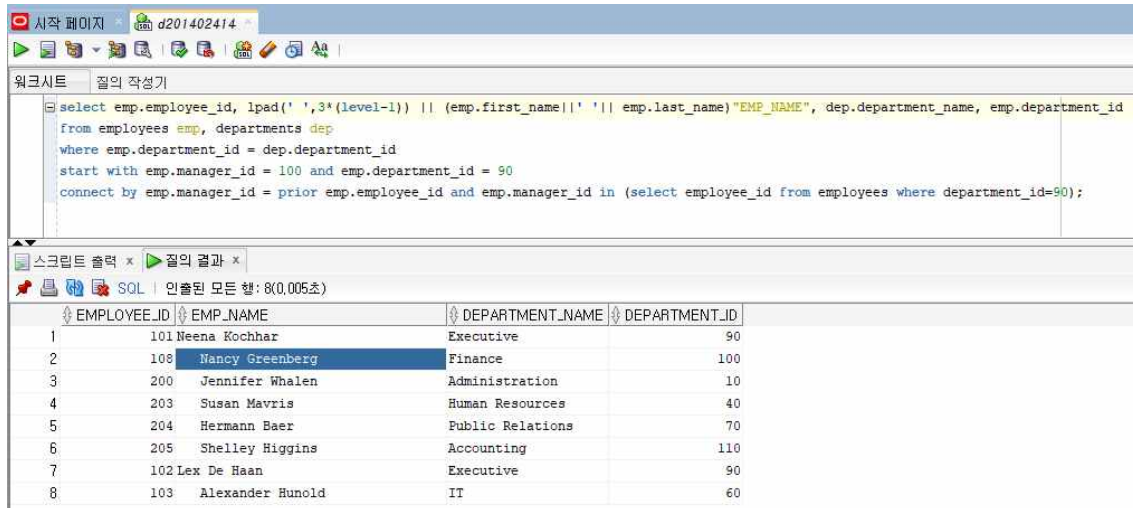
The screenshot shows a SQL IDE window with a script editor and a results pane. The script editor contains the following SQL code:

```
drop sequence Q7_01;
insert into Q7_EL
values(Q7_1.nextval, 'temp', 1100000);
```

The results pane shows an error message:

명령의 25 행에서 시작하는 중 오류 발생 -
insert into Q7_EL
values(Q7_1.nextval, 'temp', 1100000)
오류 발생 명령행: 26 열: 8
오류 보고 -
SQL 오류: ORA-02289: 시퀀스가 존재하지 않습니다.
02289. 00000 - "sequence does not exist"
*Cause: The specified sequence does not exist, or the user does not have the required privilege to perform this operation.
*Action: Make sure the sequence name is correct, and that you have the right to perform the desired operation on this sequence.

3) id가 100인 직원이 관리하는 직원들 중에 id가 90인 부서에 소속된 직원들이 직접 관리하는 직원들의 정보를 출력하시오. (단, 직원 id는 employee_id, 직원 이름은 emp_name, 부서 이름은 department_name, 부서 id는 department_id로 표시하시오.)

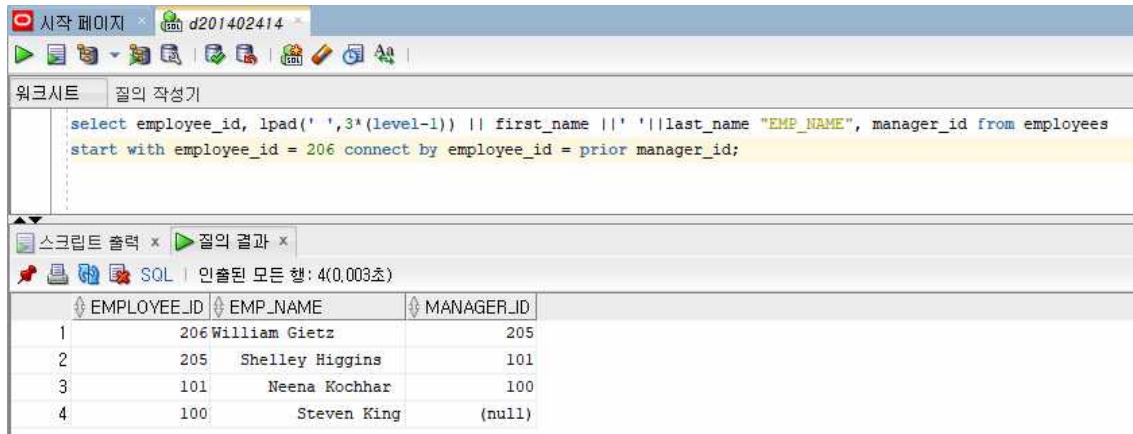


The screenshot shows a SQL IDE window with a query editor and a results pane. The query is a hierarchical query using a WITH clause to find employees managed by the employee with ID 100, specifically those in the department with ID 90. The results pane displays 8 rows of data, with the second row (Nancy Greenberg) highlighted.

```
select emp.employee_id, lpad(' ',3*(level-1)) || (emp.first_name||' '|| emp.last_name)"EMP_NAME", dep.department_name, emp.department_id
from employees emp, departments dep
where emp.department_id = dep.department_id
start with emp.manager_id = 100 and emp.department_id = 90
connect by emp.manager_id = prior emp.employee_id and emp.manager_id in (select employee_id from employees where department_id=90);
```

EMPLOYEE_ID	EMP_NAME	DEPARTMENT_NAME	DEPARTMENT_ID
1	101 Neena Kochhar	Executive	90
2	108 Nancy Greenberg	Finance	100
3	200 Jennifer Whalen	Administration	10
4	203 Susan Mavris	Human Resources	40
5	204 Hermann Baer	Public Relations	70
6	205 Shelley Higgins	Accounting	110
7	102 Lex De Haan	Executive	90
8	103 Alexander Hunold	IT	60

4) id가 206인 직원부터 상위 관리자를 찾아가는 계층 정보를 출력하시오. (단, 직원 id는 employee_id, 직원 이름은 emp_name, 관리자 id는 manager_id 로 표시하시오.)



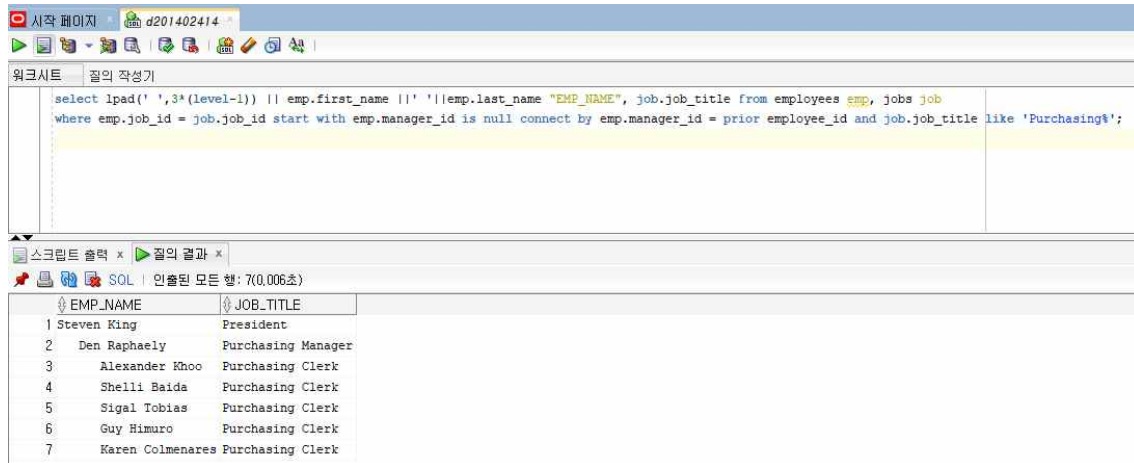
The screenshot shows a SQL IDE window with a query editor and a results pane. The query editor contains the following SQL statement:

```
select employee_id, lpad(' ', 3*(level-1)) || first_name || ' ' || last_name "EMP_NAME", manager_id from employees
start with employee_id = 206 connect by employee_id = prior manager_id;
```

The results pane displays the output of the query, showing a hierarchy of employees starting from employee_id 206. The results are as follows:

	EMPLOYEE_ID	EMP_NAME	MANAGER_ID
1	206	William Gietz	205
2	205	Shelley Higgins	101
3	101	Neena Kochhar	100
4	100	Steven King	(null)

5) 직무 유형 중에서 'Purchasing'에 관련된 직원들의 상하 계층 구조를 출력하시오. (단, 최상위 직원부터 출력하고, 직원 이름은 emp_name, 직무 유형은 job_title로 출력하시오.)



The screenshot shows a SQL IDE window with a query editor and a results pane. The query is a hierarchical query using the CONNECT BY clause to find the reporting structure for employees in the Purchasing department. The results pane shows a table with two columns: EMP_NAME and JOB_TITLE. The data is as follows:

EMP_NAME	JOB_TITLE
Steven King	President
Den Raphaely	Purchasing Manager
Alexander Khoo	Purchasing Clerk
Shelli Baida	Purchasing Clerk
Sigal Tobias	Purchasing Clerk
Guy Himuro	Purchasing Clerk
Karen Colmenares	Purchasing Clerk