

제 2 장

오라클 11g 설치 및 SQL Plus 사용법



목차

2.1 오라클 11g 설치

2.2 SQL Plus 사용법

2.3 학사관리 테이블 만들기

2.4 인사관리 데이터베이스 생성하기



SQL 오라클

- 1977년 래리 엘리슨(Larry Ellison)이 설립한 오라클사의 DBMS
- 최고의 기능성과 안정성을 갖춘 1위
- 복잡해서 입문자가 배우기는 부담스럽다.

버전	발표 시기	특징
Oracle 2.0	1979	첫 상업용 버전
Oracle 8i	1999	JVM 지원, XML 지원. i는 Internet을 의미한다.
Oracle 9i	2001	RAC(Real Application Cluster) 지원
Oracle 10g	2003	g는 여러 대의 컴퓨터를 묶은 Grid를 의미한다.
Oracle 11g	2007	엑사 데이터 지원
Oracle 12c	2013	클라우드 환경 지원. c는 Cloud를 의미한다.
Oracle 18c	2018	명령 수동 종료. APPROX_COUNT, APPROX_SUM 함수
Oracle 19c	2019	IoT를 위한 센서 데이터 저장. JSON 타입 추가
Oracle 20c	2020 예정	블록 체인 지원.

- 하이테크 관련 기술에 집중하여 DBMS 자체의 기능은 큰 변화가 없다.
- 11g 이후는 거의 비슷하다.



오라클 설치

- 개인 학습용은 무료로 설치할 수 있다.
 - <https://www.oracle.com>
- 회원 가입 후 19c 버전을 다운로드 받는다.

Download Oracle Database 19c

Get the best performance for your most demanding analytical and operational workloads.

Download Database 19c

Oracle Database 19c tutorials

Learn more about the latest innovations in Oracle Database 19c.

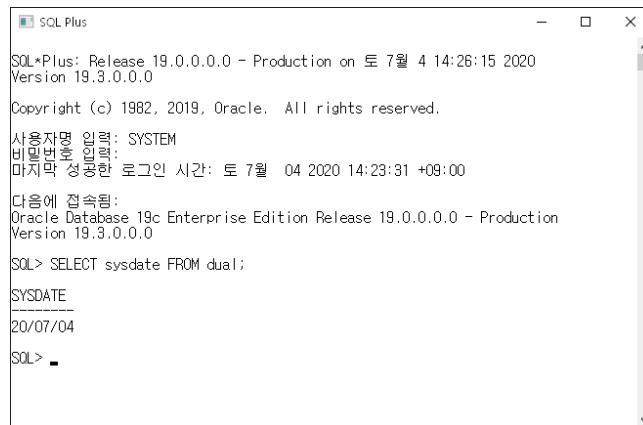
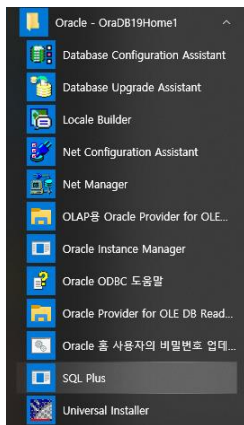
Try the tutorials

- 디폴트 옵션을 받아 들이고 기본설치의 옵션만 조정한다.



SQL Plus

- 오라클의 기본 쿼리툴이며 명령행에서 실행한다.
- 현재 시간을 출력하는 명령을 실행한다.
 - SQL>SELECT sysdate FROM dual;



- 날씨가 출력되고 SQL> 프롬프트가 다시 나타난다.
- 도스창처럼 명령을 내리면 즉시 응답하는 대화식이다.
- 기본툴 치고는 보기보다 기능이 많지만 학습용으로는 불편하다.



SQL Developer

- 오라클의 공식 쿼리툴이다. 따로 설치해야 한다.
- 검색창에서 "SQL developer"로 검색하여 다운로드 받는다.
- 압축만 풀면 바로 실행할 수 있다.
- 초록색 + 버튼을 클릭하여 접속할 서버를 등록한다.

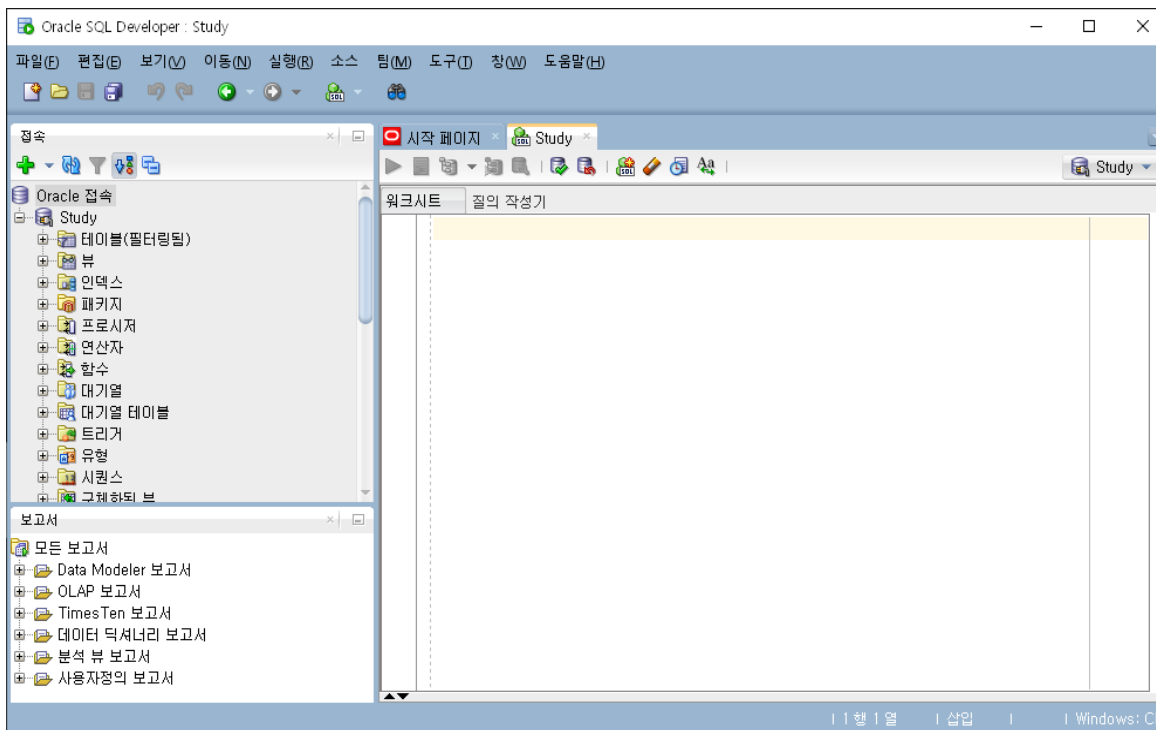
The screenshot shows the 'New/Select Database Connection' dialog box in SQL Developer. The 'Name' field is set to 'Study'. The 'Database Type' is set to 'Oracle'. The 'Username' field is set to 'SYSTEM' and the 'Password' field is masked with '****'. The 'Test' button is highlighted. The 'Host' field is set to 'localhost' and the 'Port' field is set to '1521'. The 'SID' field is set to 'Study'.

- 접속명은 Study로 지정하고 계정은 SYSTEM, 암호는 asdf로 입력한다.
- 접속 테스트 후 저장한다.



SQL Developer

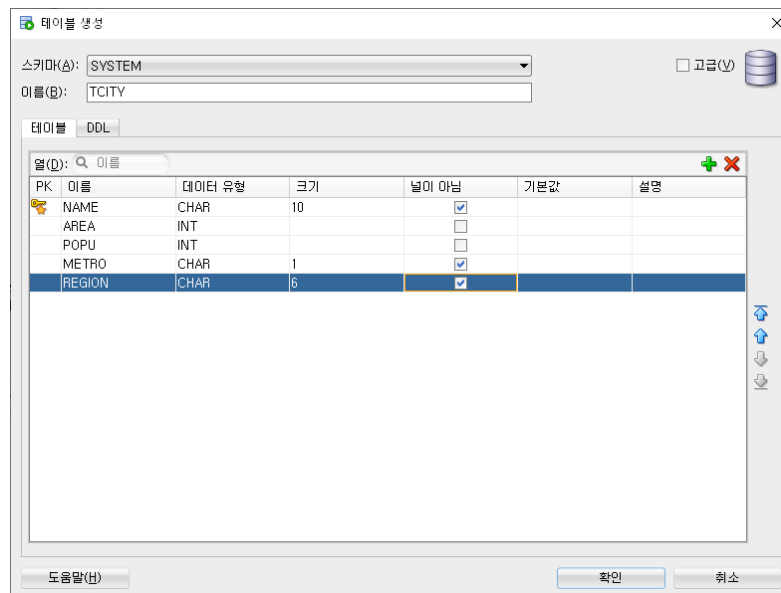
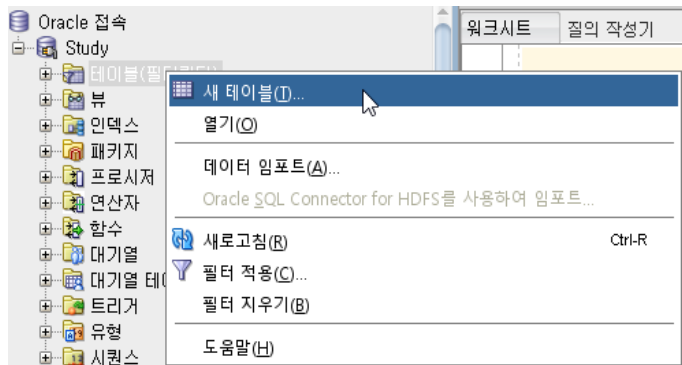
- 메뉴, 툴바, 상태란 등으로 구성된 평범한 윈도우이다.
- 접속창은 서버 목록과 서버내의 DB 오브젝트 목록이 나타난다.
- 워크시트에 쿼리 명령을 입력한다.





테이블 생성

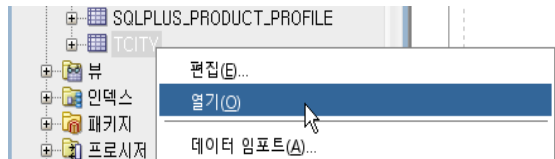
- 그래픽 환경에서도 실습을 할 수 있다.
- Study의 테이블 노드에서 팝업 메뉴를 열고 새 테이블 항목을 선택한다.
- 필드의 이름과 타입을 선택한다.
- name의 PK란을 클릭하여 기본키로 지정한다.





SQL 테이블 생성

- 테이블의 팝업 메뉴에서 열기 항목을 선택한다.
- 그리드에 데이터를 입력한다.



Study x TCITY x					
열	데이터	Model	제약 조건	권한 부여	통계
				트리거	플래시백
				종속성	사
정렬... 필터:					
	NAME	AREA	POPU	METRO	REGION
+1	서울	605	974	Y	경기
+2	부산	765	342	Y	경상
+3	오산	42	21	N	경기

- 다 입력한 후 커밋 버튼(또는 단축키 F11)을 눌러 저장한다.
- tCity는 도시 정보를 지정하는 샘플 테이블이다.

필드	타입	설명
name	CHAR(10)	도시명
area	INT	면적. 제곱킬로미터 단위
popu	INT	인구. 만명 단위
metro	CHAR(1)	광역시 여부, y 또는 n
region	CHAR(6)	소속 지역



SQL 마우스 작업의 단점


- 마우스 작업은 쉽고 직관적이지만 비효율적이다.
 - 작업 속도가 느리다.
 - 사람이 직접 수행해야 하며 복사하거나 저장할 수 없다.

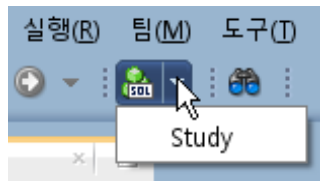
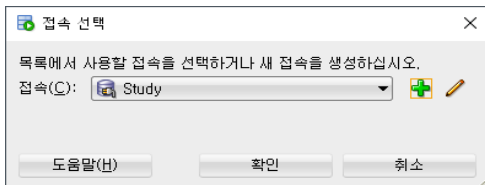


- 틀마다 사용 방법이 다르다.
- 그래픽 환경에서 데이터베이스를 관리하는 경우는 드물다.
- 스크립트는 복사 및 붙여 넣기가 가능해 저장, 전달, 재현이 쉽다.
- Copy & Paste는 개발자의 강력한 무기이다.

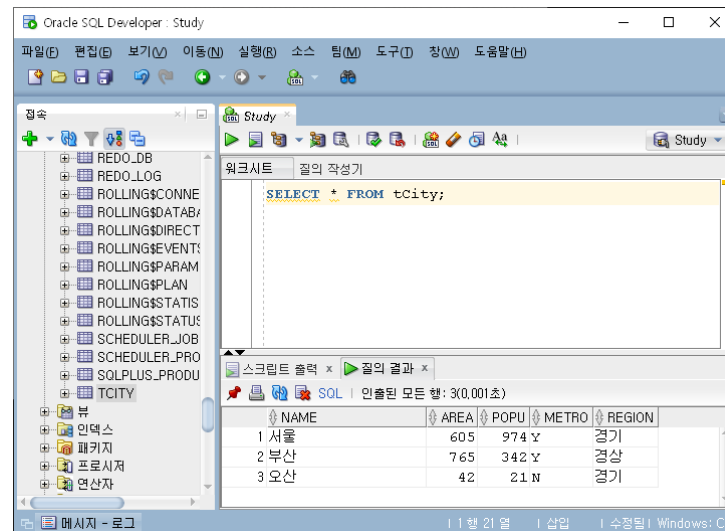


SQL 쿼리창

- 쿼리는 문자열 형태로 표현한 명령문이며 개발자와 DBMS가 통신하는 수단
- 실무에서는 대부분의 작업을 쿼리로 수행한다.
- 툴바의  버튼을 눌러 쿼리창을 열고 접속명을 선택한다.
- 툴바 버튼의 드롭 다운 버튼에서 아예 Study를 선택하면 쿼리창이 바로 열린다.



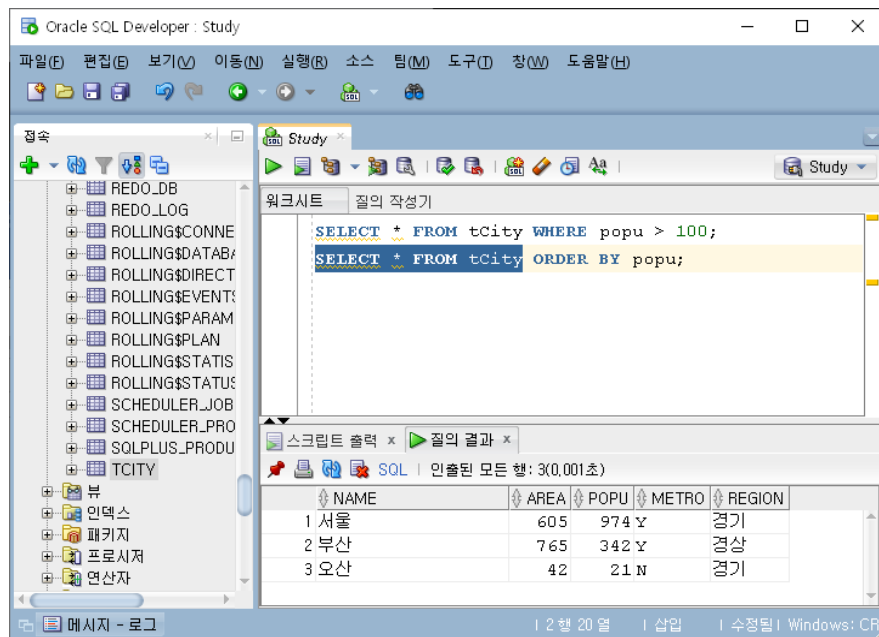
- 쿼리창에 명령을 입력하고 Ctrl+Enter를 누른다. 명령 실행 결과가 아래쪽의 질의 결과 창에 나타난다.
- 대소문자는 굳이 구분할 필요 없다.
- 모든 명령은 ;으로 끝난다.





SQL 쿼리창

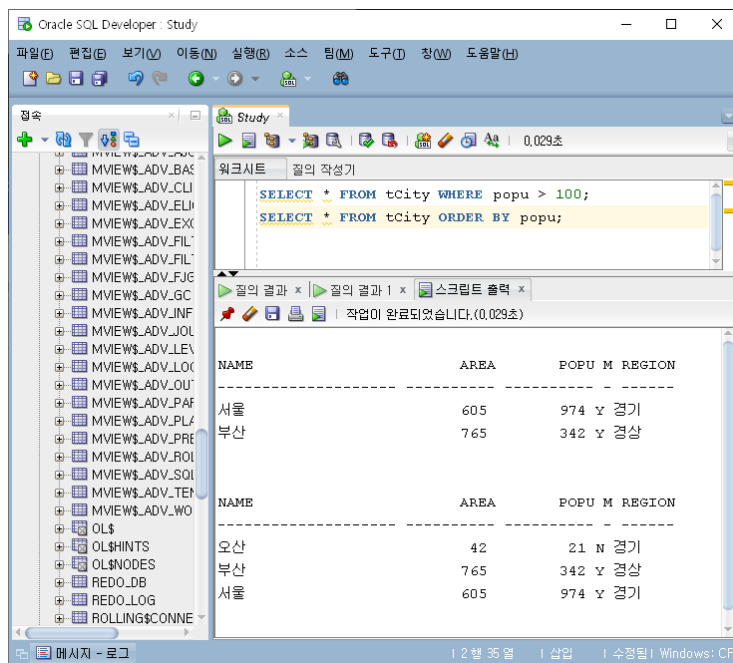
- 두 개 이상의 명령을 입력해 놓고 선택적으로 실행할 수 있다.
 - `SELECT * FROM tCity WHERE popu > 100;`
 - `SELECT * FROM tCity ORDER BY popu;`
- **Ctrl+Enter**는 캐럿이 있는 행의 명령만 실행한다.
- 블록을 선택하면 블록내의 명령만 실행한다.
- 두 개 이상의 명령은 별도의 탭을 열어 보여준다.
- 이전 쿼리를 굳이 지우지 않아도 상관 없다.





SQL 쿼리창

- F5는 스크립트를 실행한다.
- 여러 명령의 실행 결과를 한꺼번에 보여 준다.
- 실행 결과를 문자열로 출력하여 복사하기 쉽다.

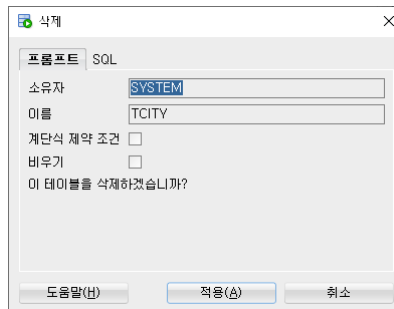
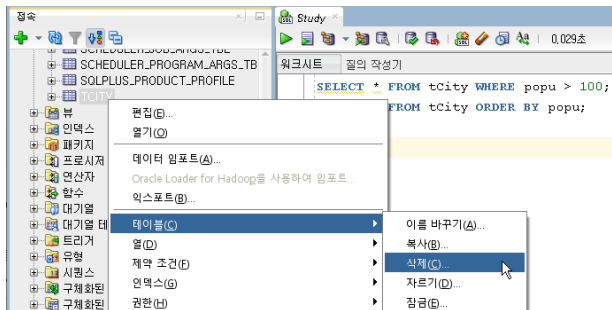


- 닫을 때 저장 여부를 묻는다. 실습용 쿼리는 굳이 저장하지 않아도 상관 없다.



SQL 쿼리문의 분류

- SQL 쿼리문은 명령의 역할에 따라 다음 3가지로 분류한다.
 - DDL(Data Definition Language) : DB 오브젝트를 생성, 삭제, 변경한다. CREATE, DROP, ALTER 등의 명령이 있다. DB를 디자인하는 관리자가 이 부류의 명령을 주로 사용한다.
 - DML(Data Manipulation Language) : DB를 조회, 삽입, 삭제, 변경한다. SELECT, INSERT, DELETE, UPDATE 명령 등이 있다. 응용 프로그램 개발자가 주로 사용한다.
 - DCL(Data Control Language) : 사용자와 권한을 관리하는 GRANT, DENY, REVOKE 등의 명령이 있다. DBA가 주로 사용하며 일반 개발자는 사용할 일이 드물다.
- 개발자는 주로 DML을 사용한다.
- 모든 DB 객체는 언제든지 삭제할 수 있다.
- 팝업 메뉴에서 삭제 항목을 선택한다.



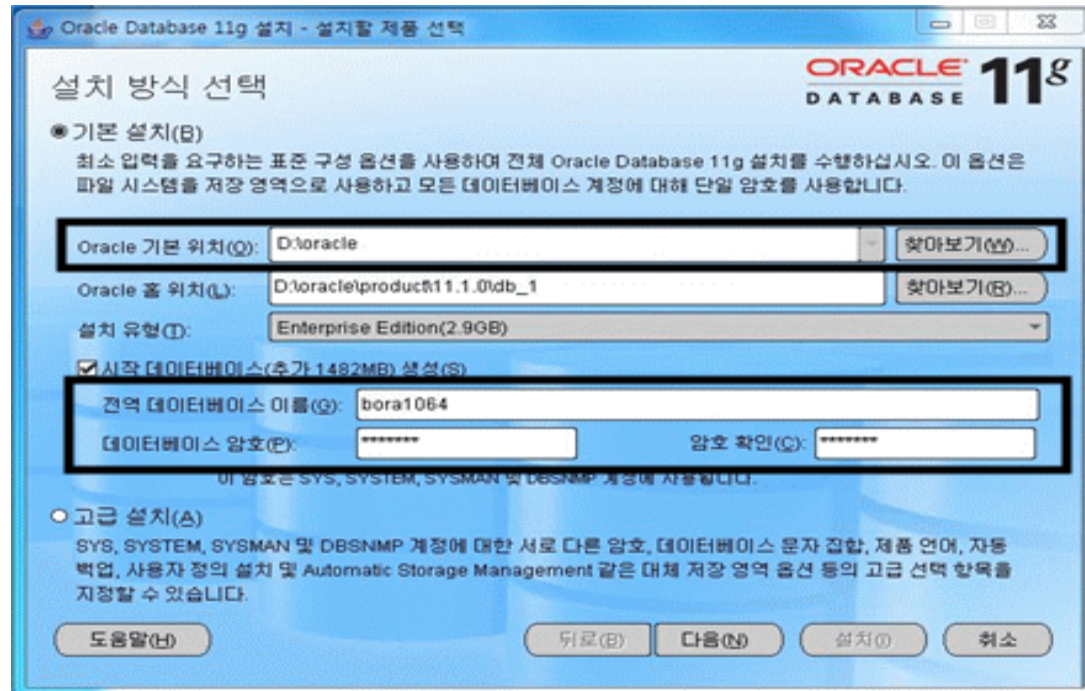
- 스크립트로 다시 만들 수 있다.

2.1 오라클 11g 설치

- <http://www.oracle.com>에서 윈도우10 기반의 오라클 11g를 1file , 2file 둘 다 다운받는다.
- 다운받은 win32_11gR1_database_1013.zip의 압축을 풀고 database 폴더에 있는 setup.exe를 실행한다.

2.1 오라클 11g 설치

- ① 오라클 기본 위치는 현재 시스템에 가장 많은 여유 공간을 가지는 드라이브로 선택해주며, 다른 드라이브에 설치하기 원한다면 [찾아보기] 버튼을 이용한다.
- ② **전역 데이터베이스 이름과 암호를 입력한다(반드시 기억할 것).** [다음] 버튼을 클릭한다.



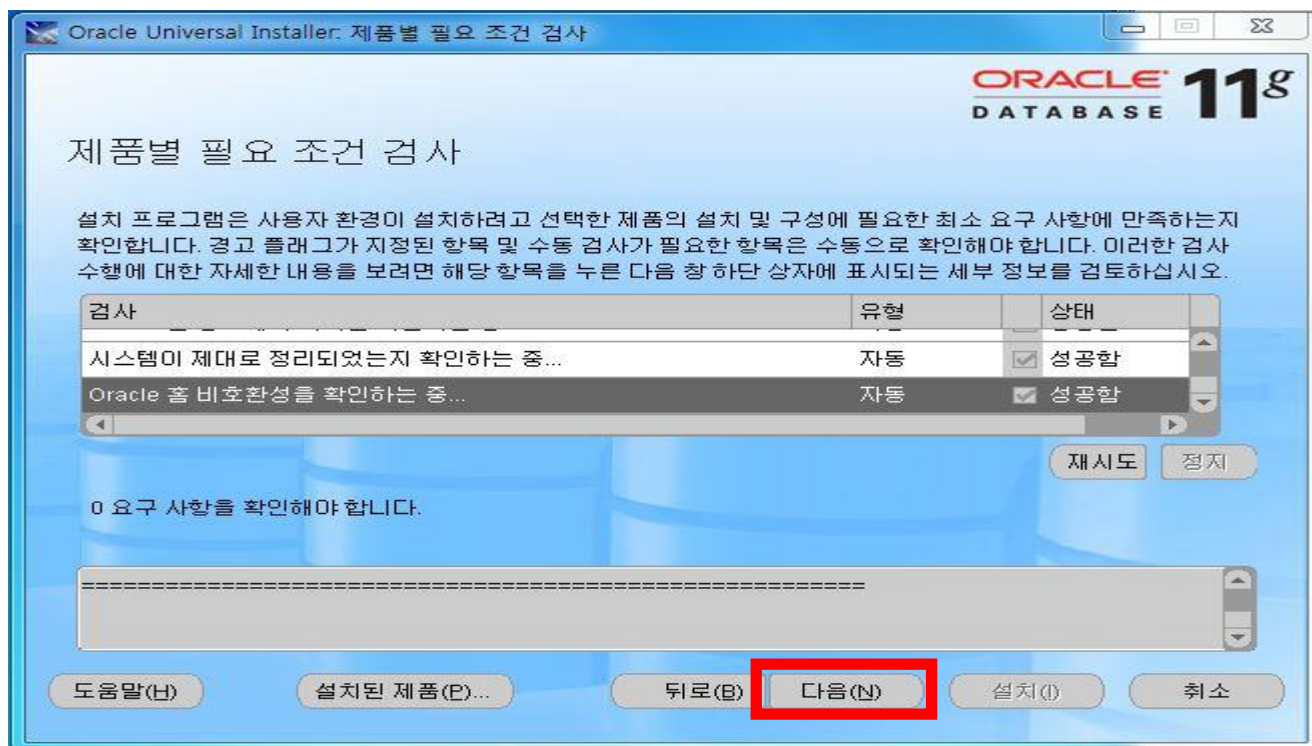
2.1 오라클 11g 설치

③ 요약 창에서 [설치] 버튼을 클릭한다.



2.1 오라클 11g 설치

- ④ 제품별 필요조건 검사 창에서 모든 검사의 상태가 성공함으로 표시되면 [다음] 버튼을 클릭한다.



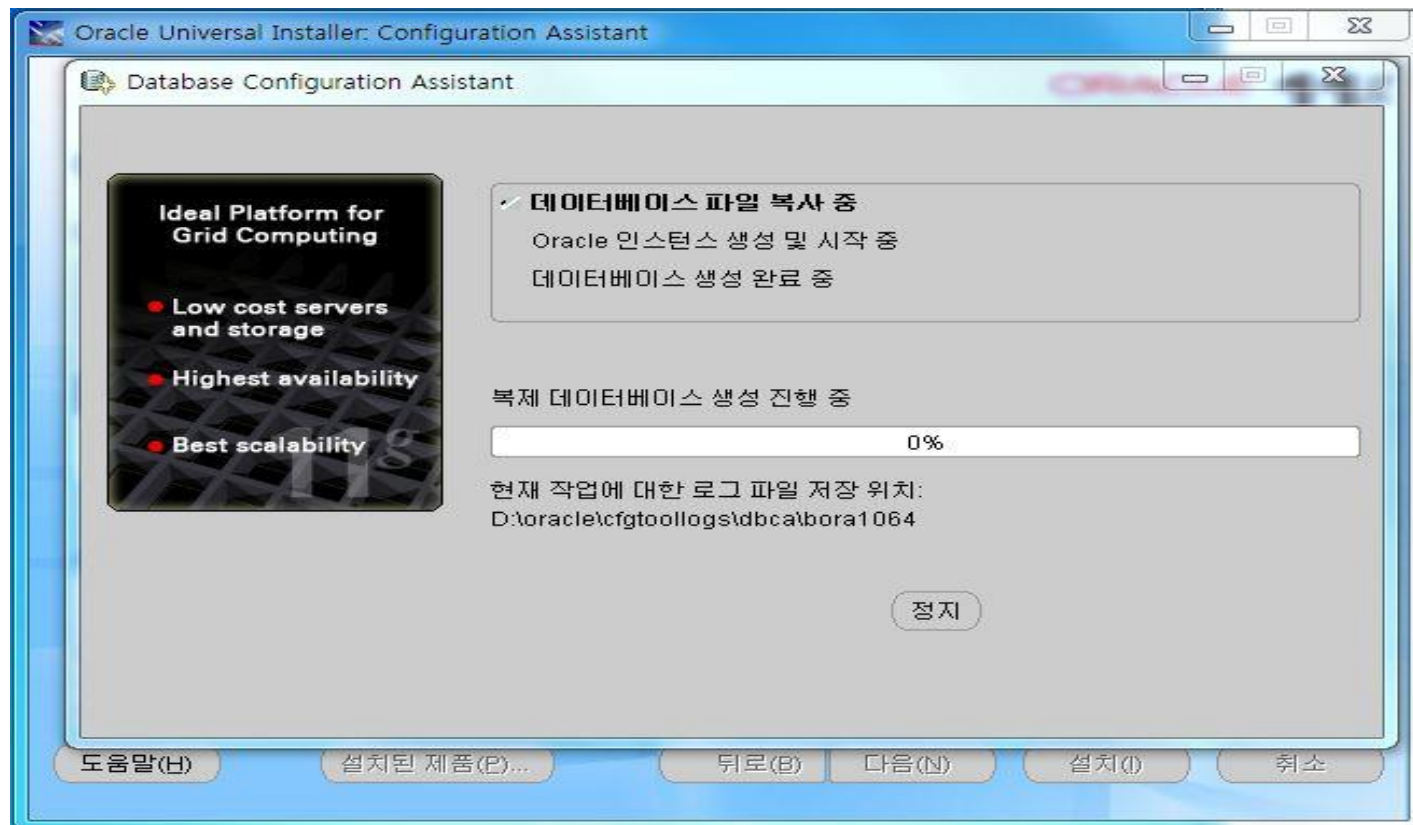
2.1 오라클 11g 설치

⑤ [다음] 버튼을 클릭한다.



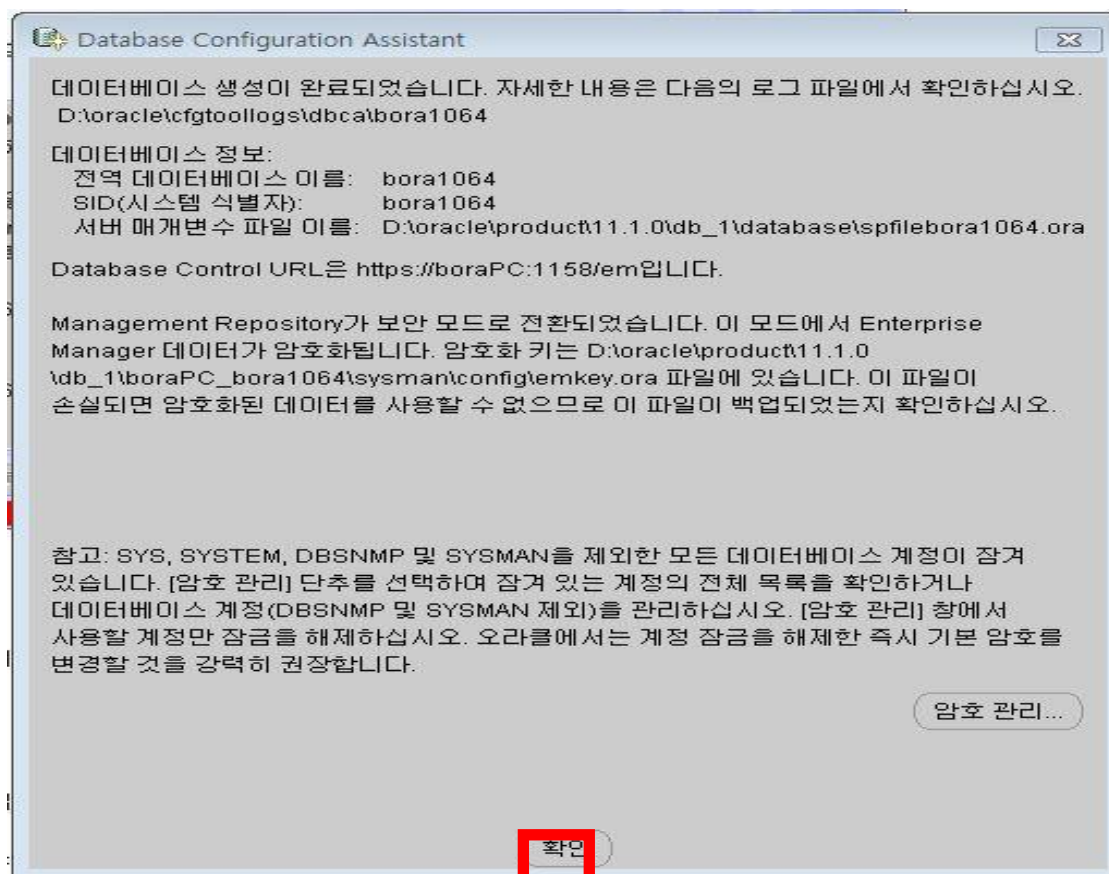
2.1 오라클 11g 설치

⑥ 데이터베이스가 설치되기를 기다린다.



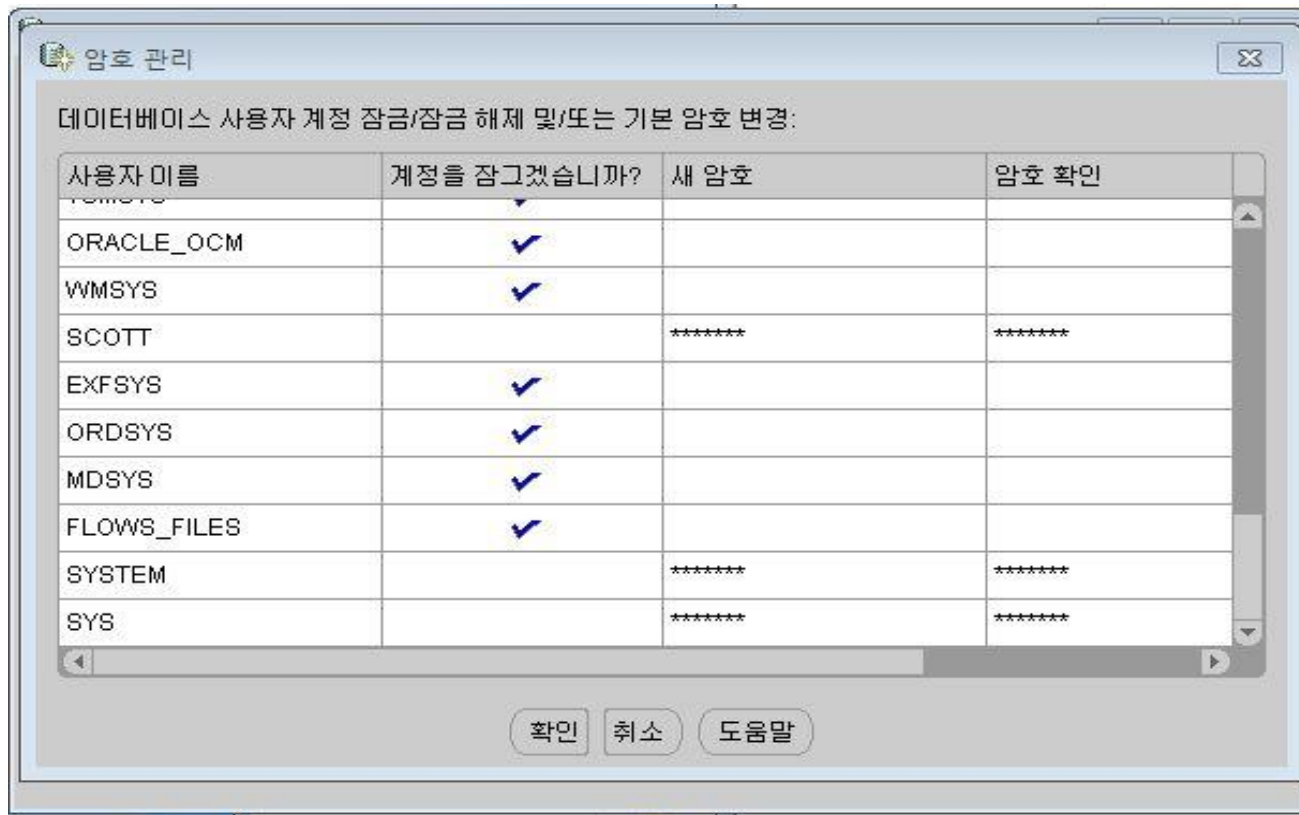
2.1 오라클 11g 설치

- ⑦ 데이터베이스 설치에 대한 로그 파일의 내용을 확인하고 [확인] 버튼을 클릭한다.



2.1 오라클 11g 설치

- ⑧ 데이터베이스 사용자 계정에 대한 화면으로 [확인] 버튼을 클릭한다.



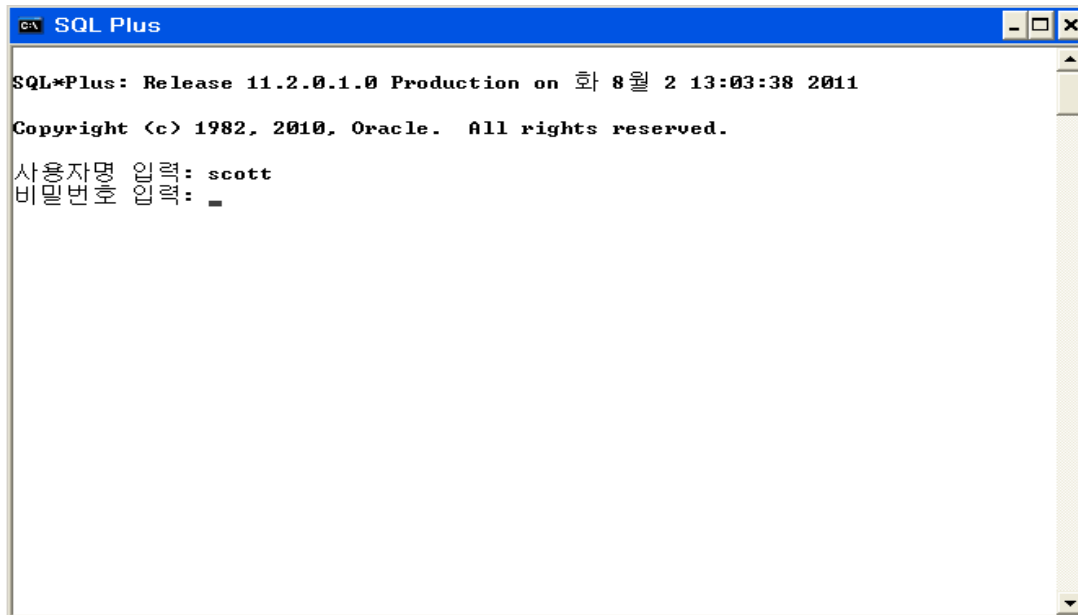
2.1 오라클 11g 설치

- ⑨ 설치가 성공적으로 완료되었다.
[종료] 버튼을 클릭하여 오라클 설치를 종료한다.



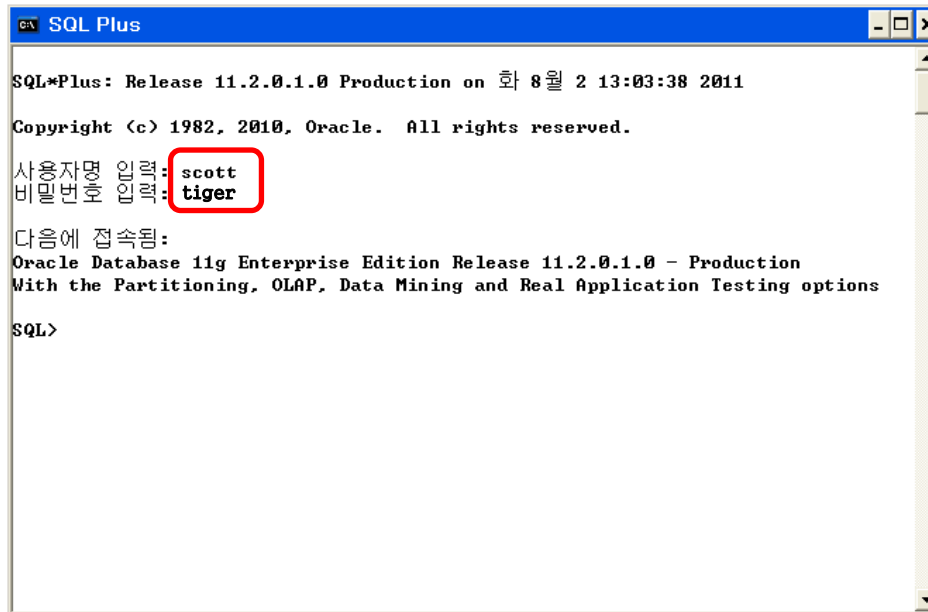
2.2 SQL Plus 사용법

- 오라클을 설치하면 기본적으로 데이터베이스에 접속하여 SQL과 PL/SQL을 실행할 수 있도록 SQL Plus툴이 설치된다.
- 윈도우 바탕화면에서 [시작] → [프로그램] → [Oracle - OraDb11g_home1] → [응용 프로그램 개발] → [SQL PLUS]를 클릭하면 다음 화면이 나타나며, **사용자명 입력 : scott, 비밀번호 입력 : tiger**를 입력하면,



2.2 SQL Plus 사용법

- 다음 화면으로 이동한다.




```
SQL*Plus: Release 11.2.0.1.0 Production on 화 8월 2 13:03:38 2011
Copyright (c) 1982, 2010, Oracle. All rights reserved.
사용자명 입력: scott
비밀번호 입력: tiger
다음에 접속됨:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL>
```

- 이제 SQL Plus를 이용하여 오라클에 대한 SQL, PL/SQL 실습을 할 수 있다.

2.2 SQL Plus 사용법

- DESCRIBE (테이블 구조보기)
- 오라클에서 제공하는 실습을 위한 사원(employees) 테이블의 구조를 다음과 같이 DESCRIBE 명령을 이용하여 확인한다.

SQL> desc employees;



```
Copyright (c) 1982, 2010, Oracle. All rights reserved.

사용자명 입력: scott
비밀번호 입력:

다음에 접속됨:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> desc emp;
이름              널?       유형
-----
EMPNO              NOT NULL  NUMBER(4)
ENAME              VARCHAR2(10)
JOB                VARCHAR2(9)
MGR                NUMBER(4)
HIREDATE           DATE
SAL                NUMBER(7,2)
COMM               NUMBER(7,2)
DEPTNO             NUMBER(2)

SQL>
```

- DESCRIBE, DESC, desc.

2.2 SQL Plus 사용법

- 다음으로 사원(employees) 테이블의 데이터를 조회해 보자.

SQL> select * from employees;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK		80/12/17	800		20
7499	ALLEN	SALESMAN	7902	81/02/20	1600	300	30
7521	WARD	SALESMAN	7698	81/02/22	1250	500	30
7566	JONES	MANAGER	7698	81/04/02	2975		20
7654	MARTIN	SALESMAN	7839	81/09/28	1250	1400	30
7698	BLAKE	MANAGER	7698	81/05/01	2850		30
7782	CLARK	MANAGER	7839	81/06/09	2450		10
7788	SCOTT	ANALYST	7839	87/04/19	3000		20
7839	KING	PRESIDENT	7566	81/11/17	5000	0	10
7844	TURNER	SALESMAN	7698	81/09/08	1500		30
7876	ADAMS	CLERK	7788	87/05/23	1100		20
7900	JAMES	CLERK	7698	81/12/03	950		30
7902	FORD	ANALYST	7566	81/12/03	3000		20
7934	MILLER	CLERK	7782	82/01/23	1300		10

14 개의 행이 선택되었습니다.

2.2 SQL Plus 사용법

●LIST

- SQL Plus **가장 최근에 실행된 SQL 문장은 버퍼에 저장하고** 있으며, 버퍼의 내용을 확인하기 위해 LIST 명령을 사용한다.

```
SQL> list
```

```
2  select *
```

```
3*  from emp
```

2.2 SQL Plus 사용법

●CHANGE

- change는 **SQL 명령을 수정하기 위한 명령어로** 위의 예에서 *표는 현재 라인을 의미하며, 현재 라인이 2번인 상태에서 다음과 같이 change/emp/dept를 사용하면

```
SQL> change/emp/dept  
2*   from dept
```

- 현재 라인 2의 emp가 dept로 수정된다. list 명령으로 확인해 보자.

```
SQL> list  
1   select *  
2*   from dept
```

2.2 SQL Plus 사용법

● R[UN]

➤ RUN은 현재 SQL 명령을 실행하는 명령어로 UN을 생략하고 R만 사용할 수 있다.

SQL> run

1 select *

2* from dept

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

4 개의 행이 선택되었습니다.

2.2 SQL Plus 사용법

● SAVE

- 현재 사용한 SQL 문장을 저장할 수 있으며, SAVE 명령을 사용한다.
- 이때 파일의 확장자는 sql이며,
- 저장위치는 오라클을 설치할 때 지정했던 홈 위치 밑에 BIN 폴더 저장된다.

SQL> save test1

file test1.sql(0)가 생성되었습니다.

2.2 SQL Plus 사용법

- **START**

➤ 다음은 저장된 파일을 실행하는 예이며, start 명령을 이용한다.

SQL> start test1

SQL> sta test1

SQL> @test1

2.2 SQL Plus 사용법

●GET

➤ GET은 저장된 파일을 다시 버퍼로 불러오는 명령이다.

```
SQL> get test1
```

```
SQL> list
```

```
1  select *
```

```
2*  from dept
```

2.2 SQL Plus 사용법

●SPOOL

- 작업 결과를 저장하기 위해 SPOOL 명령어를 사용한다.
- 현지 실행한 모든 명령과 결과를 result.lst 파일로 저장하기 위한 명령

SQL> spool result

- 다음 명령어들을 실행한다.

SQL> select * from emp;

SQL> select * from salgrade;

SQL> spool off ;

●페이지의 폭을 100으로 한다.

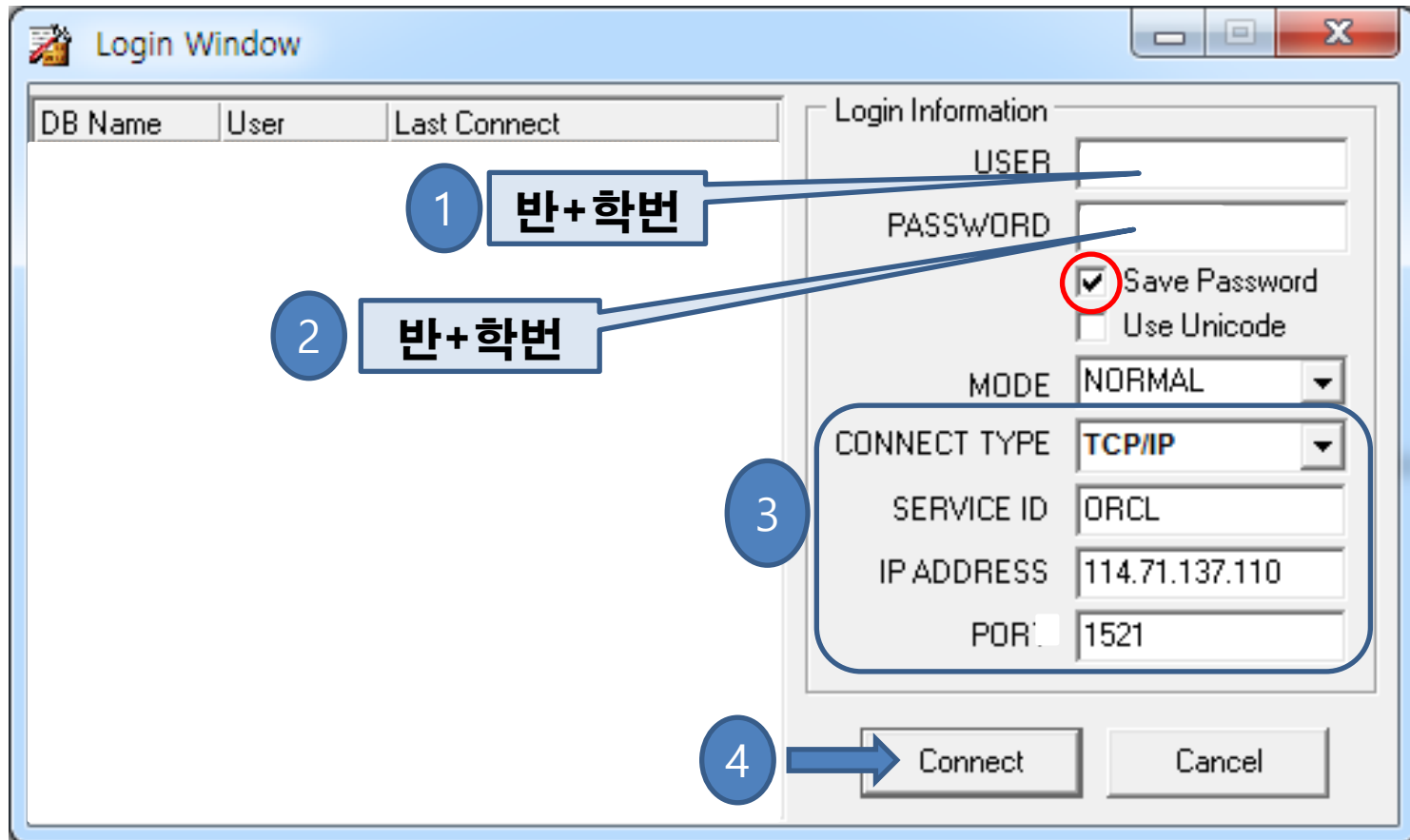
SQL> set pagesize 100;

●페이지의 결과 라인수를 50으로 한다.

SQL> set pagesize 50;

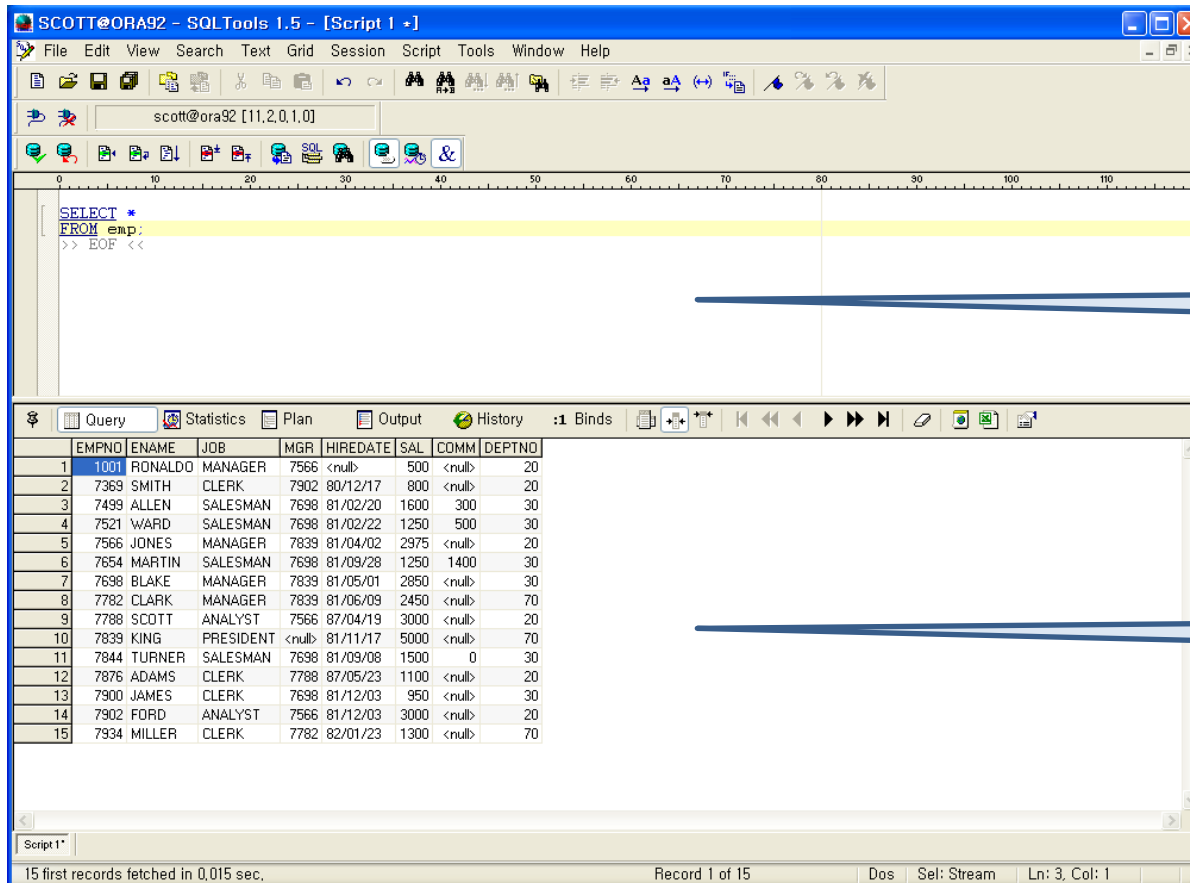
2.2 SQL Plus 사용법

●SQL Tools for Oracle



2.2 SQL Plus 사용법

●SQLTools for Oracle



SQL Editor

결과



SQL 명칭 규칙

- **DB 오브젝트는 서로 구분하기 위해 고유한 이름이 있어야 한다.**
 - 같은 범위 내에서 이름이 중복되어서는 안 된다.
 - 대소문자는 구분하지 않으므로 마음대로 붙여도 된다. 일관성을 지키는 것이 좋다.
 - 길이는 최대 128자까지 가능하다.
 - SQL 예약어는 쓸 수 없다.
 - 유니코드에 포함된 모든 문자를 다 사용할 수 있다.
 - #이나 @ 같은 기호로 시작하는 명칭은 DBMS가 정의한 특별한 의미를 가진다. 언더바(_)외의 기호는 가급적 쓰지 않는다.
- **명시적인 규칙보다 의미를 명확히 알 수 있는 간결한 이름을 붙이는 것이 더 중요하다.**
- **오브젝트의 종류나 타입을 구분하는 접두어를 붙이는 방식을 권장한다.**
 - DB : 첫 자만 대문자. Study
 - 키워드 : 모두 대문자. SELECT, CREATE
 - 테이블 : t 접두로 시작하며 첫 자는 대문자로 쓴다. tCity, tMember
 - 필드 : 모두 소문자. 여러 단어일 경우 어근만 대문자. name, orderDate
- **한 번 정한 규칙을 일관되게 지키는 것이 바람직하다.**



SQL 데이터 타입

- 정보의 크기나 형태를 규정하는 것을 데이터 타입(Data Type)이라고 한다.
 - 정보의 크기에 꼭 맞게 메모리를 알뜰하게 사용한다. 수십억을 넘어가는 화폐 액수는 큰 메모리가 필요하지만 사람의 나이는 고작 200을 넘지 않으니 약간의 메모리면 충분하다.
 - 실수나 에러로 인한 사고를 방지할 수 있다. 정수로 정의한 나이 정보에는 15, 39 같은 숫자만 들어가야지 "아직 어림", "비밀임" 따위의 문자열을 저장해서는 안된다.
 - DBMS가 타입을 미리 알고 있으면 최적화된 방법으로 정보를 읽고 빠른 속도로 연산할 수 있어 성능이 향상된다.
- 5개의 타입만 해도 현실 세계의 데이터 98% 이상을 표현할 수 있다.

타입	설명
INT	정수
DECIMAL	실수
CHAR	고정 길이 문자열
VARCHAR	가변 길이 문자열
DATE	날짜



데이터 타입

- 고정 길이인 CHAR 타입은 길이보다 짧은 문자열의 뒷부분을 공백으로 채우는데 비해 가변 길이인 VARCHAR 타입은 문자열 길이만큼만 저장한다.

abc
한글
SQL Server

CHAR(10)

3	abc
4	한글
10	SQL Server

VARCHAR(10)

- VARCHAR형은 용량면에서 유리하고 CHAR형은 길이가 일정해 속도가 빠르다.
- 길이가 일정한 필드에는 CHAR형을 쓰는 것이 좋고 길이가 들쭉날쭉한 문자열은 VARCHAR 타입이 적합하다.

21	좋은 정보 감사합니다.
3	^ _ ^
37	덕분에 골치아픈 문제 잘 해결했습니다.
23	금전 대출. 015-396-1523

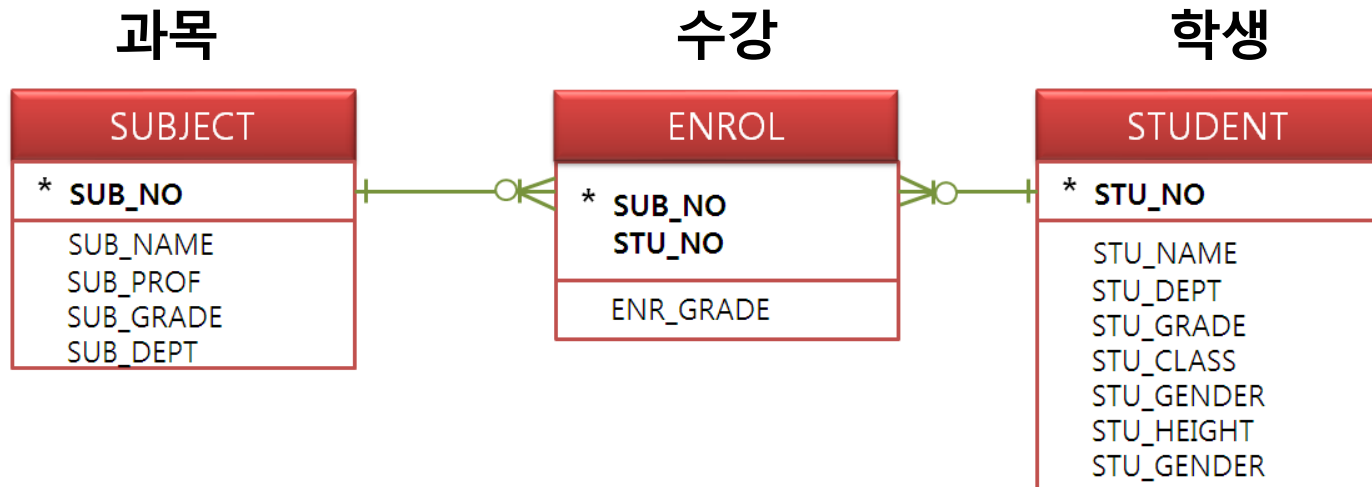
VARCHAR(100)

좋은 정보 감사합니다.
^ _ ^
덕분에 골치아픈 문제 잘 해결했습니다.
금전 대출. 015-396-1523

CHAR(100)

2.3 학사관리 테이블 만들기

● 학사관리의 3개 테이블에 대한 ERD



2.3 학사관리 테이블 만들기(학생테이블)

SQL> select * from student;

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20153075	옥한빛	기계	1	C	M	177	80
20153088	이태연	기계	1	C	F	162	50
20143054	유가인	기계	2	C	F	154	47
20152088	조민우	전기전자	1	C	M	188	90
20142021	심수정	전기전자	2	A	F	168	45
20132003	박희철	전기전자	3	B	M		63
20151062	김인중	컴퓨터정보	1	B	M	166	67
20141007	진현무	컴퓨터정보	2	A	M	174	64
20131001	김종헌	컴퓨터정보	3	C	M		72
20131025	옥성우	컴퓨터정보	3	A	F	172	63

2.3 학사관리 테이블 만들기(과목테이블)

SQL> select * from subject;

SUB_NO	SUB_NAME	SUB_PROFESSOR	SUB_GRADE	SUB_DEPT
111	데이터베이스	이재영	2	컴퓨터정보
110	자동제어	정순정	2	전기전자
109	자동화설계	박민영	3	기계
101	컴퓨터개론	강종영	3	컴퓨터정보
102	기계공작법	김태영	1	기계
103	기초전자실험	김유석	1	전기전자
104	시스템분석설계	강석현	3	컴퓨터정보
105	기계요소설계	김명성	1	기계
106	전자회로실험	최영민	3	전기전자
107	CAD응용실습	구봉규	2	기계
108	소프트웨어공학	권민성	1	컴퓨터정보

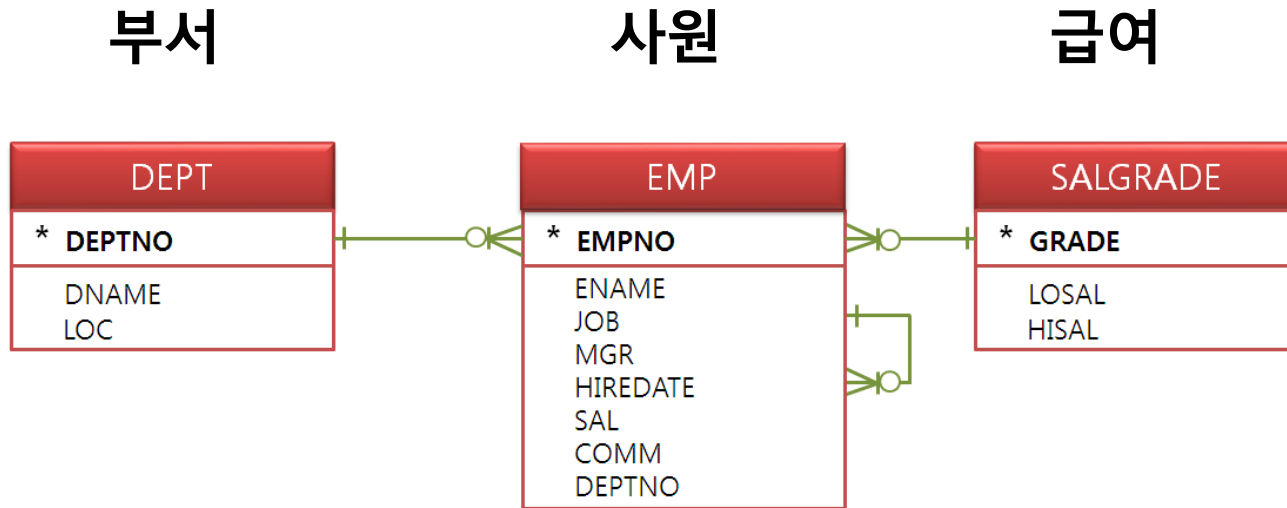
2.3 학사관리 테이블 만들기(수강테이블)

SQL> select * from enrol;

SUB_NO	STU_NO	ENR_GRADE
101	20131001	80
104	20131001	56
106	20132003	72
103	20152088	45
101	20131025	65
104	20131025	65
108	20151062	81
107	20153054	41
102	20153075	66
105	20153075	56
102	20153088	61
105	20153088	78

2.4 인사관리 데이터베이스 생성하기

●인사관리에 3개 테이블에 대한 ERD



2.4 인사관리 데이터베이스 생성하기

●3개 테이블에 대한 명세서

➤사원 테이블

Table ID		EMP	Tabel Des.		사원	
Col. name	Explain	Data type	length	Null	Key	
EMPNO	사원번호	char	4	X	O	
ENAME	사원이름	Varchar2	30	X		
JOB	사원직책	Varchar2	20	X		
MGR	상사사원번호	Char	4	O		
HIREDATE	입사일	Date		X		
SAL	급여	number	7,2	X		
COMM	커미션	number	7,2	O		
DEPTNO	부서번호	char	2	X		

사원 테이블

```
create table emp(  
    empno number(4)    -- 사원번호  
    constraint pk_emp primary key, --PK 선언  
    ename varchar2(10), --사원이름  
    job varchar2(9),   -- 직무  
    mgr number(4),     -- 매니저(상급자)  
    hiredate date,     -- 입사일자  
    sal number(7,2),   -- 연봉  
    comm number(7,2),  -- 보너스(상여금)  
    deptno number(2)   -- 부서번호  
);
```

```
INSERT INTO EMP VALUES (7839,'KING','PRESIDENT',NULL,to_date('17-11-1981','dd-mm-yyyy'),5000,NULL,10);
INSERT INTO EMP VALUES (7566,'JONES','MANAGER',7839,to_date('2-4-1981','dd-mm-yyyy'),2975,NULL,20);
INSERT INTO EMP VALUES (7698,'BLAKE','MANAGER',7839,to_date('1-5-1981','dd-mm-yyyy'),2850,NULL,30);
INSERT INTO EMP VALUES (7782,'CLARK','MANAGER',7839,to_date('9-6-1981','dd-mm-yyyy'),2450,NULL,10);
INSERT INTO EMP VALUES (7788,'SCOTT','ANALYST',7566,to_date('13-07-1987','dd-mm-yyyy'),3000,NULL,20);
INSERT INTO EMP VALUES (7902,'FORD','ANALYST',7566,to_date('3-12-1981','dd-mm-yyyy'),3000,NULL,20);
INSERT INTO EMP VALUES (7499,'ALLEN','SALESMAN',7698,to_date('20-2-1981','dd-mm-yyyy'),1600,300,30);
INSERT INTO EMP VALUES (7521,'WARD','SALESMAN',7698,to_date('22-2-1981','dd-mm-yyyy'),1250,500,30);
INSERT INTO EMP VALUES (7654,'MARTIN','SALESMAN',7698,to_date('28-9-1981','dd-mm-yyyy'),1250,1400,30);
INSERT INTO EMP VALUES (7844,'TURNER','SALESMAN',7698,to_date('8-9-1981','dd-mm-yyyy'),1500,0,30);
INSERT INTO EMP VALUES (7900,'JAMES','CLERK',7698,to_date('3-12-1981','dd-mm-yyyy'), 950,NULL,30);
INSERT INTO EMP VALUES (7934,'MILLER','CLERK',7782,to_date('23-1-1982','dd-mm-yyyy'),1300,NULL,10);
INSERT INTO EMP VALUES (7369,'SMITH','CLERK',7902,to_date('17-12-1980','dd-mm-yyyy'), 800,NULL,20);
INSERT INTO EMP VALUES (7876,'ADAMS','CLERK',7788,to_date('13-07-1987','dd-mm-yyyy'),1100,NULL,20);
```

2.4 인사관리 데이터베이스 생성하기

● 3개 테이블에 대한 명세서

➤ 부서 테이블

Table ID		DEPT	Tabel Des.	부서	
Col. name	Explain	Data type	length	Null	Key
DEPTN0	부서번호	char	2	X	O
DNAME	부서이름	varchar2	14	X	
LOC	주소	varchar2	13	X	

➤ 급여등급 테이블

Table ID		SALGRADE	Tabel Des.	급여 등급	
Col. name	Explain	Data type	length	Null	Key
GRADE	급여등급	char	2	X	O
LOSAL	최저급여	number	7,2	X	
HISAL	최고급여	number	7,2	X	

부서 테이블

CREATE TABLE DEPT

**(DEPTNO NUMBER(2) CONSTRAINT PK_DEPT PRIMARY KEY, --부서번호
DNAME VARCHAR2(14), --부서이름
LOC VARCHAR2(13)); --부서위치**

INSERT INTO DEPT VALUES (10,'ACCOUNTING','NEW YORK');

INSERT INTO DEPT VALUES (20,'RESEARCH','DALLAS');

INSERT INTO DEPT VALUES (30,'SALES','CHICAGO');

INSERT INTO DEPT VALUES (40,'OPERATIONS','BOSTON');

급여 테이블

CREATE TABLE salgrade

**(GRADE NUMBER(7,2), --급여
LOSAL NUMBER(7,2), --하위급여
HISAL NUMBER(7,2)); --상위급여**

INSERT INTO salgrade VALUES(1,700,1200);

INSERT INTO salgrade VALUES(2,1201,1400);

INSERT INTO salgrade VALUES(3,1401,2000);

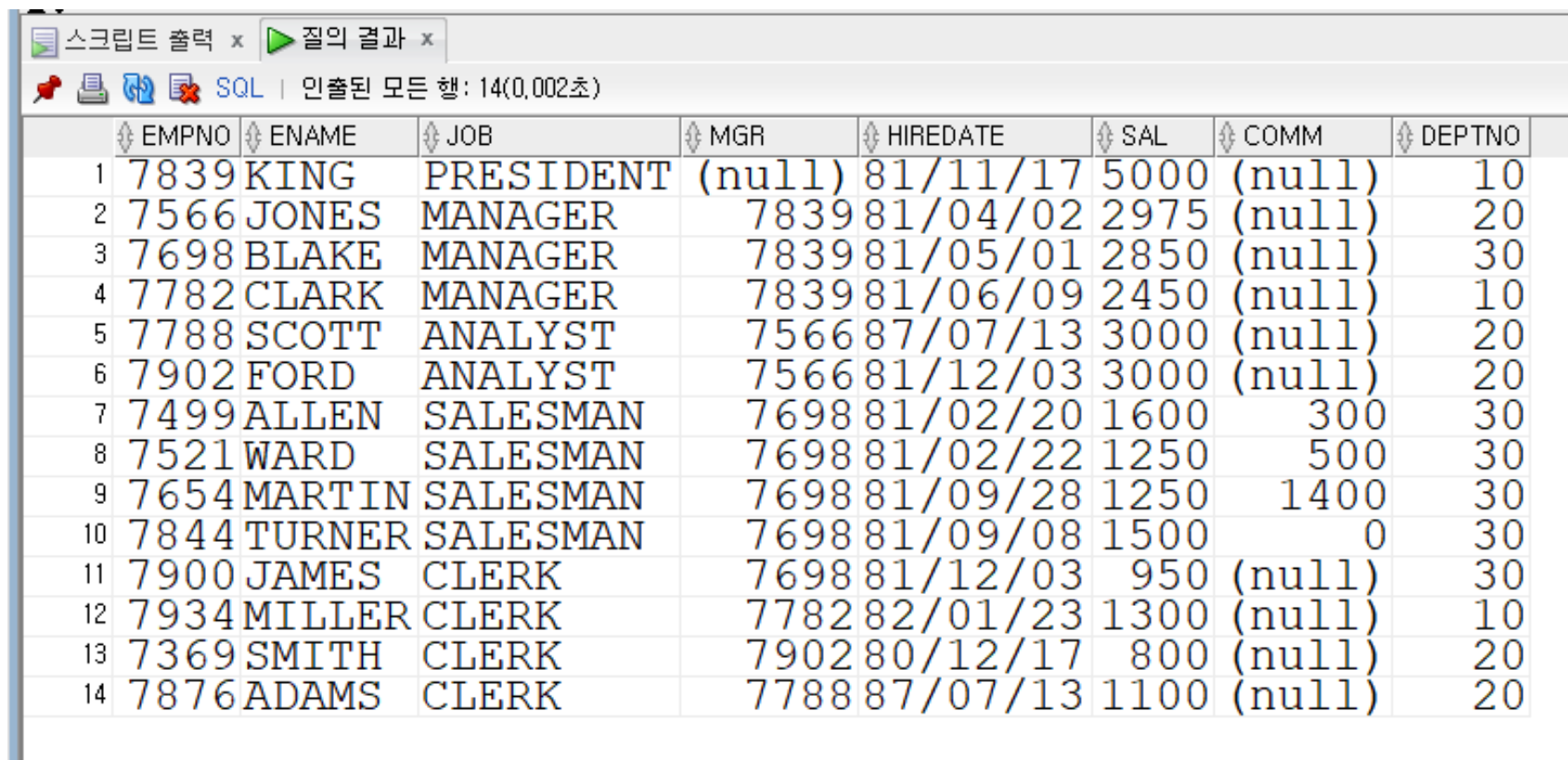
INSERT INTO salgrade VALUES(4,2001,3000);

INSERT INTO salgrade VALUES(5,3001,9999);

2.4 인사관리 데이터베이스 생성하기

- 각 테이블에 데이터를 데이터를 확인한다.

SQL> select * from emp;



	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7839	KING	PRESIDENT	(null)	81/11/17	5000	(null)	10
2	7566	JONES	MANAGER	7839	81/04/02	2975	(null)	20
3	7698	BLAKE	MANAGER	7839	81/05/01	2850	(null)	30
4	7782	CLARK	MANAGER	7839	81/06/09	2450	(null)	10
5	7788	SCOTT	ANALYST	7566	87/07/13	3000	(null)	20
6	7902	FORD	ANALYST	7566	81/12/03	3000	(null)	20
7	7499	ALLEN	SALESMAN	7698	81/02/20	1600	300	30
8	7521	WARD	SALESMAN	7698	81/02/22	1250	500	30
9	7654	MARTIN	SALESMAN	7698	81/09/28	1250	1400	30
10	7844	TURNER	SALESMAN	7698	81/09/08	1500	0	30
11	7900	JAMES	CLERK	7698	81/12/03	950	(null)	30
12	7934	MILLER	CLERK	7782	82/01/23	1300	(null)	10
13	7369	SMITH	CLERK	7902	80/12/17	800	(null)	20
14	7876	ADAMS	CLERK	7788	87/07/13	1100	(null)	20

2.4 인사관리 데이터베이스 생성하기

SQL> select * from dept;

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

2.4 인사관리 데이터베이스 생성하기

SQL> select * from salgrade;

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

2장을 마치며.....

Q & A