



1장 데이터베이스 개념과 오라클 설치

- 유용한 데이터의 집합
- 데이터베이스에 저장된 정보는 검색, 수정, 삭제에 용이

1.1 데이터베이스와 데이터베이스 관리시스템

1.1.1 파일 시스템의 문제점

- 파일시스템에 다음과 같은 문제점이 제기되면서 새로운 시스템(DBMS)에 대한 요구가 제기되었다.

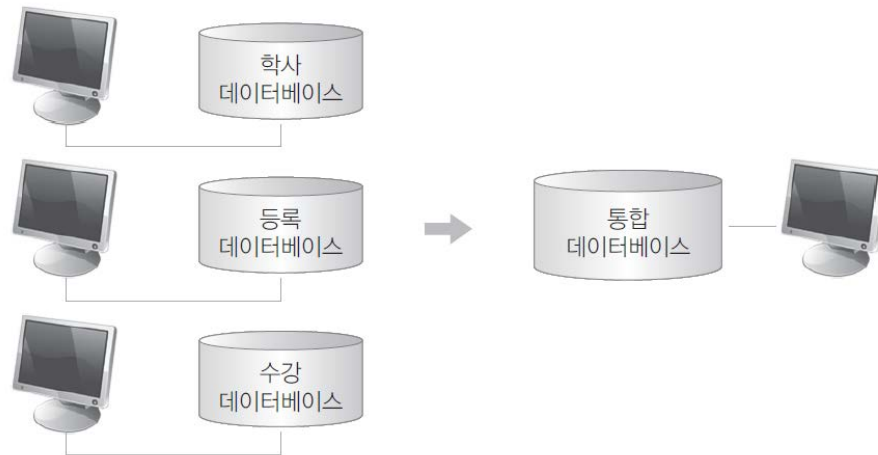
구분	파일 시스템	DBMS
데이터 중복	데이터를 파일 단위로 저장하므로 중복 가능	DBMS를 이용하여 데이터를 공유하기 때문에 중복 가능성 낮음
데이터 일관성	데이터의 중복 저장으로 일관성이 결여됨	중복 제거로 데이터의 일관성이 유지됨
데이터 독립성	데이터 정의와 프로그램의 독립성 유지 불가능	데이터 정의와 프로그램의 독립성 유지 가능
관리 기능	보통	데이터 복구, 보안, 동시성 제어, 데이터 관리 기능 등을 수행
프로그램 개발 생산성	나쁨	짧은 시간에 큰 프로그램을 개발할 수 있음
기타 장점	보통	데이터 무결성 유지, 데이터 표준 준수 용이

[표] 파일 시스템과 DBMS의 비교

1.1.2 데이터베이스의 개념

- 데이터베이스의 개념은 다음과 같이 네 가지로 설명할 수 있다.
 - 통합된 데이터(integrated data)
 - 데이터를 통합하는 개념으로, 각자 사용하던 데이터의 중복을 최소화하여 중복으로 인한 데이터 불일치 현상을 제거
 - 저장된 데이터(stored data)
 - 문서로 보관된 데이터가 아니라 디스크, 테이프 같은 컴퓨터 저장장치에 저장된 데이터를 의미
 - 운영 데이터(operational data)
 - 조직의 목적을 위해 사용되는 데이터를 의미한다. 즉 업무를 위한 검색을 할 목적으로 저장된 데이터
 - 공용 데이터(shared data)

- 한 사람 또는 한 업무를 위해 사용되는 데이터가 아니라 **공동**으로 사용되는 데이터를 의미



1.1.3 데이터베이스의 특징

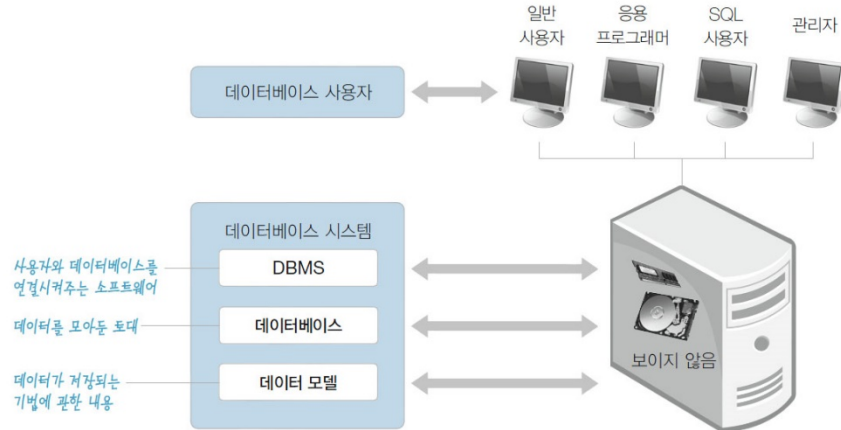
- 데이터베이스의 특징은 다음과 같이 네 가지로 설명할 수 있다.
 - 실시간 접근성(real time accessibility)
 - 데이터베이스는 실시간으로 서비스된다. 사용자가 데이터를 요청하면 몇 시간이나 몇 일 뒤에 결과를 전송하는 것이 아니라 수 초 내에 결과를 서비스한다.
 - 지속적인 변화(continuous change)
 - 데이터베이스에 저장된 내용은 어느 한 순간의 상태를 나타내지만, 데이터 값은 시간에 따라 항상 바뀐다. 데이터베이스는 삽입(insert), 삭제(delete), 수정(update) 등의 작업을 통하여 바뀐 데이터 값을 저장한다.
 - 동시 공유(concurrent sharing)
 - 데이터베이스는 서로 다른 업무 또는 여러 사용자에게 동시에 공유된다. 동시(concurrent)는 병행이라고도 하며, 데이터베이스에 접근하는 프로그램이 여러 개 있다는 의미다.
 - 내용에 따른 참조(reference by content)
 - 데이터베이스에 저장된 데이터는 데이터의 물리적인 위치가 아니라 데이터 값에 따라 참조된다

1.1.4 데이터베이스 시스템의 구성

- 데이터베이스 시스템은 각 조직에서 사용하던 데이터를 모아서 통합하고 서로 공유하여 생기는 장점을 이용하는 시스템이다.
- 데이터베이스(DataBase)
 - 기업이 지속적으로 유지 관리해야 하는 데이터의 집합
- 데이터베이스 관리 시스템(DataBase Management System)
 - 방대한 양의 데이터를 편리하게 저장하고 효율적으로 관리하고 검색할 수 있는 환경을 제공해주는 시스템 소프트웨어
 - 데이터를 공유하여 정보의 체계적인 활용을 가능하게 한다.
 - 응용 프로그램과 데이터베이스의 중재자로서 모든 응용 프로그램들이 데이터베이스를 공유할 수 있게끔 관리해 주는 소프트웨어 시스템이다.

■ 데이터 모델

- 데이터가 저장되는 기법에 관한 내용, 예) 가전제품에 붙여진 모델 번호가 제품의 특성을 나타내듯이 데이터 모델은 데이터가 저장되는 스타일을 나타낸다.



[그림] 데이터베이스 시스템의 구성요소

1.1.5 관계형 데이터베이스 관리 시스템

- 관계형 데이터베이스 관리시스템(RDBMS: Relational DataBase Management System)은 가장 일반적인 형태의 DBMS 이다.
- 관계형 데이터베이스 정보를 테이블 형태로 저장한다. 테이블은 2차원 형태의 표처럼 볼 수 있도록 로우(ROW:행)와 칼럼(COLUMN:열)으로 구성한다.

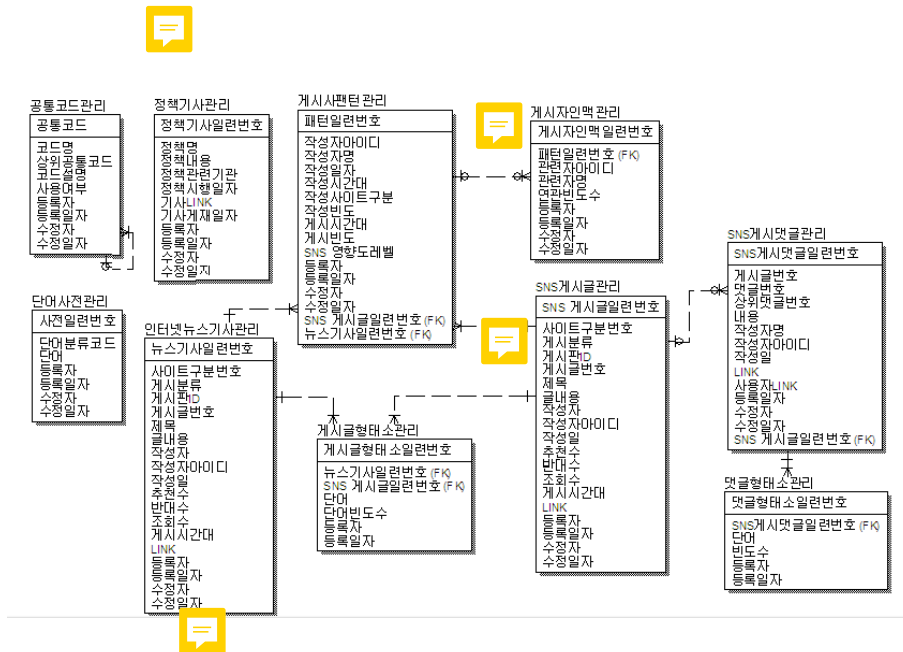


테이블 이름 : DEPT

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



- ERD(Entity Relationship Diagram)



- RDBMS의 종류: Oracle, MS SQL Server, DB2, Sybase, Infomix, MySQL, MS Access, Tibero, Postgres, MongoDB, MariaDB, NoSQL 등이다.

[꿀팁] DB-Engines Ranking (<https://db-engines.com/en/ranking>)

348 systems in ranking, November 2018

Rank			DBMS	Database Model	Score		
Nov 2018	Oct 2018	Nov 2017			Nov 2018	Oct 2018	Nov 2017
1.	1.	1.	Oracle +	Relational DBMS	1301.11	-18.16	-58.94
2.	2.	2.	MySQL +	Relational DBMS	1159.89	-18.22	-162.14
3.	3.	3.	Microsoft SQL Server +	Relational DBMS	1051.55	-6.78	-163.53
4.	4.	4.	PostgreSQL +	Relational DBMS	440.24	+20.85	+60.33
5.	5.	5.	MongoDB +	Document store	369.48	+6.30	+39.01
6.	6.	6.	IBM Db2 +	Relational DBMS	179.87	+0.19	-14.19
7.	7.	9.	Redis +	Key-value store	144.17	-1.12	+22.99
8.	8.	10.	Elasticsearch +	Search engine	143.46	+1.13	+24.05
9.	9.	7.	Microsoft Access	Relational DBMS	138.44	+1.64	+5.12
10.	11.	11.	SQLite +	Relational DBMS	122.71	+5.96	+9.95
11.	10.	8.	Cassandra +	Wide column store	121.74	-1.64	-2.47
12.	13.	15.	Splunk	Search engine	80.37	+3.48	+15.50
13.	12.	12.	Teradata +	Relational DBMS	79.31	+0.67	+1.07
14.	14.	18.	MariaDB +	Relational DBMS	73.25	+0.12	+17.96
15.	16.	19.	Hive +	Relational DBMS	64.57	+3.47	+11.32
16.	15.	13.	Solr	Search engine	60.87	-0.44	-8.28
17.	17.	16.	HBase +	Wide column store	60.41	-0.26	-3.15
18.	18.	14.	SAP Adaptive Server +	Relational DBMS	56.57	-2.00	-10.47

1.2 오라클 설치 및 설정

1.2.1 오라클 설치

- OTN 가입 : <http://www.oracle.com/kr>
- Oracle Database XE 11g Release 2 다운로드
 - <http://www.oracle.com/technology/software/products/database/index.html>
- Oracle Database XE 11g Release 2 설치
 - 전역 데이터베이스 이름(SID): XE
 - 데이터베이스 암호: "sys"로 설정한다.

- 설치 디렉토리: **C:\prod\oraclexe**
- 주요 오라클 서비스 (Microsoft Windows 버전인 경우)
- OracleServiceXE : OracleService + SID명 형태로 구성된 서비스로 설치한 오라클 기본 서비스이며 오라클 사용 시 반드시 맨 먼저 시작되어야 한다.
 - OracleXETNSListener : 리스너 서비스로 이것 역시 반드시 시작되어야 한다. 리스너란 네트워크를 통해 클라이언트(오라클을 사용하려는 사용자)와 오라클 서버와의 연결을 담당하는 관리 프로그램이다.

[꿀팁] 오라클 네트워크 접속 설정

```
// 1. d:\oracle\app\product\11.2.0\dbhome_1\NETWORK\ADMIN\listener.ora 파일 설정
# listener.ora Network Configuration File: D:\oracle\app\product\11.2.0\dbhome_1\network\admin\listener.ora
# Generated by Oracle configuration tools.
```

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = CLRExtProc)
      (ORACLE_HOME = D:\oracle\app\product\11.2.0\dbhome_1)
      (PROGRAM = extproc)
      (ENVS = "EXTPROC_DLLS=ONLY:D:\oracle\app\product\11.2.0\dbhome_1\bin\oraclr11.dll")
    )
  )
```

```
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    )
  )
```

```
ADR_BASE_LISTENER = D:\oracle\app
```

```
// 2. d:\oracle\app\product\11.2.0\dbhome_1\NETWORK\ADMIN\tnsnames.ora 파일 설정
# tnsnames.ora Network Configuration File: D:\oracle\app\product\11.2.0\dbhome_1\network\admin\tnsnames.ora
# Generated by Oracle configuration tools.
```

```
LISTENER_MYORACLE =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
```

```
ORACLR_CONNECTION_DATA =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
    (CONNECT_DATA =
      (SID = CLRExtProc)
      (PRESENTATION = RO)
    )
  )
```

```
MYORACLE =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
```

```

        (SERVICE_NAME = myoracle)
    )
)

ORCL =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = myoracle)
  )
)

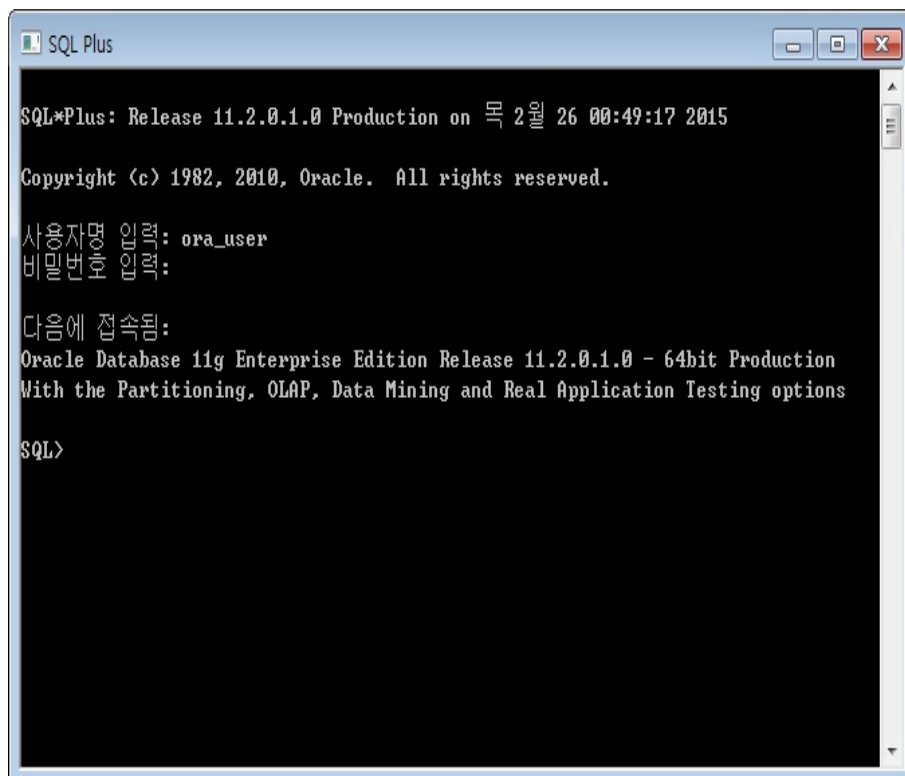
```

1.2.2 개발 도구 소개

- 오라클 설치 후 SQL이나 PL/SQL을 실행하고 결과를 보려면 개발도구가 필요함
- 대표적인 개발 도구로는 SQL Plus, 토드, 오렌지, PL/SQL Developer, SQL Developer 등이 있음

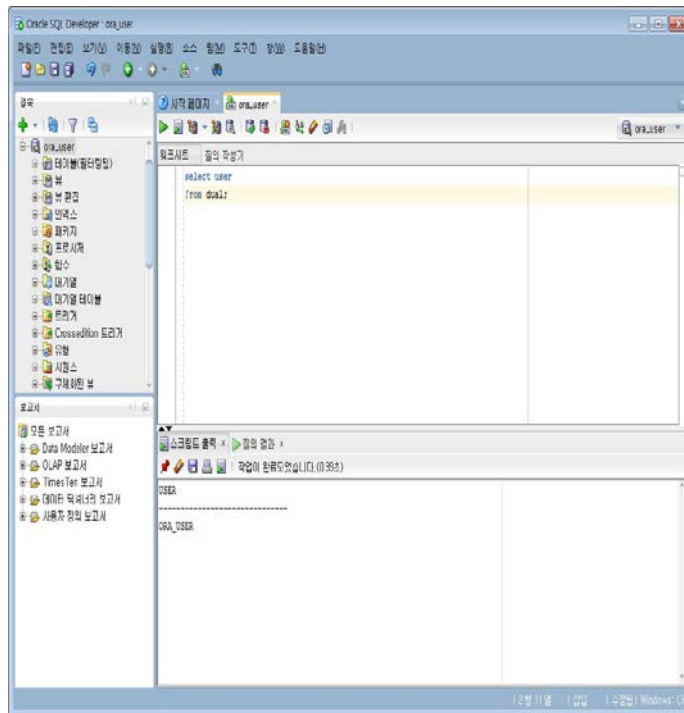
(1) SQLPLUS

- 오라클에 내장된 툴로 윈도우의 명령창 화면과 유사
- SQL, PL/SQL 뿐만 아니라 오라클 내부적인 명령어도 사용 가능



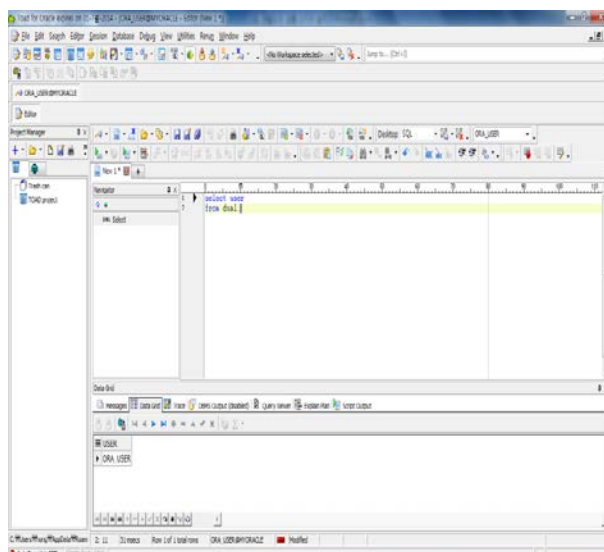
(2) SQL Developer

- 오라클에서 제공하는 무료 개발도구
- 다운로드: <http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html>



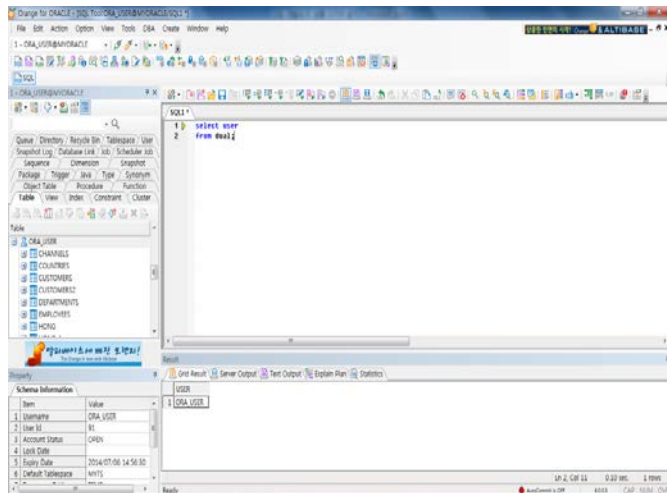
(3) Toad

- 전 세계적으로 개발자들 사이에 가장 많이 사용됨
- 상용제품이나 프리웨어 버전도 존재
- <http://www.toadworld.com>



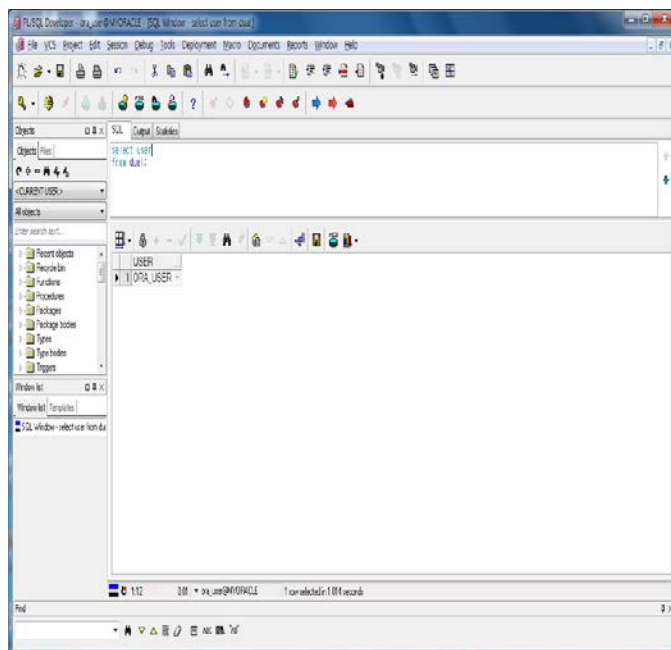
(4) Orange

- 국산 제품으로 화면은 토드와 비슷하고 점차 사용자가 늘고 있다.
- 상용제품으로 프리웨어 버전은 없고 기간제한의 Trial 버전은 존재
- <http://www.warevalley.com/>



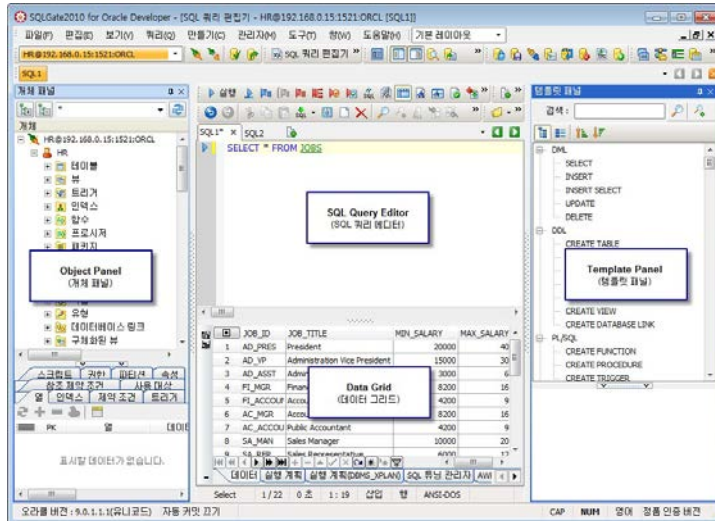
(5) PL/SQL Developer

- 토드나 오렌지와 GUI 면에서는 차이가 있지만, 나름 개발하기에 편리한 툴
- 상용제품으로 프리웨어 버전은 없고 기간제한의 Trial 버전은 존재
- <http://www.allroundautomations.com>



(6) SQLGate for Oracle Developer

- 테이블 설계 및 정의서 작성에 용이하다.
- 다운로드: <https://www.sqlgate.com/>



1.2.3 샘플 스키마

(1) SCOTT계정 활성화

```
C:\TEMP> sqlplus system/sys
SQL> CREATE USER SCOTT IDENTIFIED BY TIGER;
SQL> GRANT CONNECT, RESOURCE TO SCOTT;
SQL> @SCOTT.SQL
SQL> CONN SCOTT/TIGER
SQL> SHOW USER
SQL> SELECT COUNT(*) FROM TAB;
SQL> QUIT;
```

(2) 오라클 접속

```
C:\TEMP> SQLPLUS SCOTT/TIGER
```

(3) 문제해결

- 계정이 잠금되었을때

```
SQL> CONN SCOTT/TIGER
ERROR:
ORA-28000: 계정이 잠금되었습니다
경고: 이제는 ORACLE에 연결되어 있지 않습니다.
```

```
SQL> ALTER USER SCOTT ACCOUNT UNLOCK;
```

사용자가 변경되었습니다.

```
SQL> CONN SCOTT/TIGER
연결되었습니다.
SQL>
```

■ DBA 및 USER 비밀번호 까먹었을때

```
SQLPLUS /"AS SYSDBA"
접속 후
ALTER USER ID IDENTIFIED BY PW;
EX) ALTER USER SYS IDENTIFIED BY ORA123;

CF.
오라클 SYS, SYSTEM 암호 까먹었을때
명령 프롬프트에서 다음을 실행합니다.

C:>SQLPLUS "/AS SYSDBA"
SQL> SHOW USER
USER IS "SYS"

암호를 원하는 대로 설정합니다.

SQL> ALTER USER SYS IDENTIFIED BY 암호;
SQL> ALTER USER SYSTEM IDENTIFIED BY 암호;
```

1.2.4 과제 스키마

(1) 과제 스키마 설치

- ch01_script.sql, expall.dmp, expcust.dmp, export_sales.sql 파일을 다운로드 받아 'C:\Temp\oracle\01\' 폴더 생성 후 붙여 놓는다.
- 명령 창을 열어 'C:\Temp\oracle\01\' 폴더로 이동한다.

```
C:\temp\oracle\01> sqlplus system/sys
SQL> @ch01_script.sql
SQL> CONN ORA_USER/HONG
연결되었습니다.
SQL> SHOW USER
USER은 "ORA_USER"입니다
SQL> EXIT
C:\temp\oracle\01> IMP ORA_USER/HONG FILE=EXPALL.DMP LOG=EMPALL.LOG IGNORE=Y GRANTS=Y ROWS=Y INDEXES=Y FULL=Y

Import: Release 11.2.0.1.0 - Production on 일 6월 11 00:38:37 2017
...
임포트가 경고 없이 정상적으로 종료되었습니다.

C:\TEMP> IMP ORA_USER/HONG FILE=EXPCUST.DMP LOG=EMPALL.LOG IGNORE=Y GRANTS=Y ROWS=Y INDEXES=Y FULL=Y

Import: Release 11.2.0.1.0 - Production on 일 6월 11 00:38:54 2017
...
임포트가 경고 없이 정상적으로 종료되었습니다.

SQL> @export_sales.sql
```

(2) 과제 스키마 설치 확인

- SQL Plus나 SQL Developer에 접속 해 다음 명령어 실행

```
C:\temp\oracle\01> SQLPLUS ORA_USER/HONG

SQL*Plus: Release 11.2.0.1.0 Production on 일 6월 11 00:42:50 2017

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

다음에 접속됨:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SELECT TABLE_NAME FROM USER_TABLES;

TABLE_NAME
-----
SALES
PRODUCTS
KOR_LOAN_STATUS
JOBS
JOB_HISTORY
EMPLOYEES
DEPARTMENTS
CUSTOMERS
COUNTRIES
CHANNELS

10 개의 행이 선택되었습니다.

SQL>
```

(3) 과제 스키마 설명

- employees : 사원테이블 (사원번호, 사원명, 부서번호 등)
- departments : 부서테이블 (부서번호, 부서명 등)
- jobs : job 테이블 (job번호, 명칭 등)
- job_history : job_history 테이블 (job번호, 사원번호, 부서번호 등)
- countries : 국가 테이블 (국가번호, 국가코드, 국가명 등)
- customers : 고객 테이블 (고객번호, 고객명, 국가번호 등)
- channels : 판매채널 테이블(채널번호, 채널명 등)
- products : 제품 테이블 (제품번호, 제품명 등)
- sales : 판매 테이블(제품번호, 고객번호, 채널번호, 사원번호 등)

1.2.5 추가 스키마 (마당서점)

```
-- 이름: DEMO_MADANG.SQL
-- 설명
-- MADANG 스키마를 생성하고 MADANG 서점 실습테이블과 데이터를 입력한다.
-- 본스크립트는 SYSTEM 계정에서 실행해야한다.
----
-- SQLPLUS 실행방법
```

```

-- SQL>@DEMO_MADANG.SQL
-- SQL DEVELOPER F5 스크립터 실행

D:\temp\sql>SQLPLUS SYSTEM/MANAGER
SQL> ED DEMO_MADANG.SQL
DROP USER MADANG CASCADE;

CREATE USER MADANG IDENTIFIED BY MADANG DEFAULT TABLESPACE USERS TEMPORARY TABLESPACE TEMP PROFILE DEFAULT;

GRANT CONNECT, RESOURCE TO MADANG;
GRANT CREATE VIEW, CREATE SYNONYM TO MADANG;

ALTER USER MADANG ACCOUNT UNLOCK;

CONN MADANG/MADANG;

CREATE TABLE BOOK (
  BOOKID      NUMBER(2) PRIMARY KEY,
  BOOKNAME    VARCHAR2(40),
  PUBLISHER   VARCHAR2(40),
  PRICE       NUMBER(8)
);

CREATE TABLE CUSTOMER (
  CUSTID      NUMBER(2) PRIMARY KEY,
  NAME        VARCHAR2(40),
  ADDRESS     VARCHAR2(50),
  PHONE       VARCHAR2(20)
);

CREATE TABLE ORDERS (
  ORDERID     NUMBER(2) PRIMARY KEY,
  CUSTID      NUMBER(2) REFERENCES CUSTOMER(CUSTID),
  BOOKID      NUMBER(2) REFERENCES BOOK(BOOKID),
  SALEPRICE   NUMBER(8) ,
  ORDERDATE   DATE
);
-- BOOK, CUSTOMER, ORDERS 데이터 생성
INSERT INTO BOOK VALUES(1, '축구의 역사', '굿스포츠', 7000);
INSERT INTO BOOK VALUES(2, '축구하는 여자', '나무수', 13000);
INSERT INTO BOOK VALUES(3, '축구의 이해', '대한미디어', 22000);
INSERT INTO BOOK VALUES(4, '골프 바이블', '대한미디어', 35000);
INSERT INTO BOOK VALUES(5, '피겨 교본', '굿스포츠', 8000);
INSERT INTO BOOK VALUES(6, '역도 단계별기술', '굿스포츠', 6000);
INSERT INTO BOOK VALUES(7, '야구의 추억', '이상미디어', 20000);
INSERT INTO BOOK VALUES(8, '야구를 부탁해', '이상미디어', 13000);
INSERT INTO BOOK VALUES(9, '올림픽 이야기', '삼성당', 7500);
INSERT INTO BOOK VALUES(10, 'OLYMPIC CHAMPIONS', 'PEARSON', 13000);

INSERT INTO CUSTOMER VALUES (1, '박지성', '영국 맨체스터', '000-5000-0001');
INSERT INTO CUSTOMER VALUES (2, '김연아', '대한민국 서울', '000-6000-0001');
INSERT INTO CUSTOMER VALUES (3, '장미란', '대한민국 강원도', '000-7000-0001');
INSERT INTO CUSTOMER VALUES (4, '추신수', '미국 클리블랜드', '000-8000-0001');
INSERT INTO CUSTOMER VALUES (5, '박세리', '대한민국 대전', NULL);

-- 주문(ORDERS) 테이블의 책값은 할인 판매를 가정함
INSERT INTO ORDERS VALUES (1, 1, 1, 6000, TO_DATE('2014-07-01','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (2, 1, 3, 21000, TO_DATE('2014-07-03','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (3, 2, 5, 8000, TO_DATE('2014-07-03','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (4, 3, 6, 6000, TO_DATE('2014-07-04','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (5, 4, 7, 20000, TO_DATE('2014-07-05','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (6, 1, 2, 12000, TO_DATE('2014-07-07','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (7, 4, 8, 13000, TO_DATE('2014-07-07','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (8, 3, 10, 12000, TO_DATE('2014-07-08','YYYY-MM-DD'));

```

```

INSERT INTO ORDERS VALUES (9, 2, 10, 7000, TO_DATE('2014-07-09','YYYY-MM-DD'));
INSERT INTO ORDERS VALUES (10, 3, 8, 13000, TO_DATE('2014-07-10','YYYY-MM-DD'));

-- 여기는 3장에서 사용되는 IMPORTED_BOOK 테이블

CREATE TABLE IMPORTED_BOOK (
  BOOKID      NUMBER ,
  BOOKNAME    VARCHAR(40),
  PUBLISHER   VARCHAR(40),
  PRICE       NUMBER(8)
);
INSERT INTO IMPORTED_BOOK VALUES(21, 'ZEN GOLF', 'PEARSON', 12000);
INSERT INTO IMPORTED_BOOK VALUES(22, 'SOCCER SKILLS', 'HUMAN KINETICS', 15000);

COMMIT;

SQL> @DEMO_MADANG.SQL

사용자가 삭제되었습니다.
...
SQL>

```

1.3 SQL과 SQL*Plus의 개념

1.3.1 사용 용도에 따른 SQL

- SQL(Structured Query Language) : 데이터베이스에 저장된 데이터를 조회, 입력, 수정 삭제 하는 등의 조작이나, 테이블을 비롯한 다양한 객체(시퀀스, 인덱스 등)를 생성 및 제어하는 역할을 한다.
- SQL의 종류
 - 데이터 정의어(DDL) : 데이터베이스 관리자나 응용 프로그래머가 데이터베이스의 논리적 구조를 정의하기 위한 언어로서 데이터 사전(Data Dictionary)에 저장 된다.
 - 데이터 조작어(DML) : 데이터베이스에 저장된 데이터를 조작하기 위해 사용하는 언어로서 데이터 검색(Retrieval), 추가(Insert), 삭제>Delete), 갱신(Update) 작업 수행한다.
 - 데이터 제어어(DCL) : 데이터에 대한 접근 권한 부여 등의 데이터베이스 시스템의 트랜잭션을 관리하기 위한 목적으로 사용되는 언어이다.

유형	명령문
DQL:Data Query Language(질의어)	SELECT(데이터 검색시 사용)
DML:Data Manipulation Language (데이터 조작어) - 데이터 변경시 사용	INSERT(데이터 입력) UPDATE(데이터 수정) DELETE(데이터 삭제)
DDL:Data Definition Language (데이터 정의어) - 객체 생성 및 변경시 사용	CREATE(데이터베이스 생성) ALTER(데이터베이스 변경) DROP(데이터베이스 삭제) RENAME(데이터베이스 객체이름 변경) TRUNCATE(데이터베이스 저장 공간 삭제)
TCL:Transaction Control Language (트랜잭션 처리어)	COMMIT(트랜잭션의 정상적인 종료처리) ROLLBACK(트랜잭션 취소) SAVEPOINT(트랜잭션내에 임시 저장점 설정)
DCL:Data Control Language (데이터 제어어)	GRANT(데이터베이스에 대한 일련의 권한 부여) REVOKE(데이터베이스에 대한 일련의 권한 취소)



1.3.2 가장 기본이 되는 SELECT문

- SELECT문은 테이블에 저장된 데이터를 조회하는 데 사용되는 가장 기본적인 문법이고 가장 많이 쓰이는 문장이다.
 - SELECT : 데이터 검색시 사용

```
SELECT * FROM DEPT;
```

1.3.3 DML(Data Manipulation Language)

- DML은 데이터를 조작하는 역할을 한다. 새로운 데이터를 삽입하고, 기존의 데이터를 변경하고 삭제하는 것이 DML에 속한다.
 - INSERT : 데이터 입력
 - UPDATE : 데이터 수정
 - DELETE : 데이터 삭제

```
INSERT INTO DEPT VALUES(50, '총무부', '서울');  
UPDATE DEPT SET LOC='부산' WHERE DNAME='총무부';  
DELETE FROM DEPT WHERE DEPTNO=50;
```

1.3.4 TCL(Transaction Control Language)

- TCL은 데이터의 일관성을 유지하면서 안정적으로 데이터를 복구시키기 위해서 사용한다.
 - COMMIT : 변경된 내용을 영구 저장한다.
 - ROLLBACK : 변경되기 이전 상태로 되돌린다.
 - SAVEPOINT : 특정 위치까지를 저장 혹은 이전 상태로 되돌리 수 있도록 트랜잭션 중에 저장점을 만든다.

1.3.5 DDL(Data Definition Language)

- DDL은 데이터베이스 객체들을 생성 또는 변경, 제거할 때 사용한다. 객체란 테이블, 인덱스, 뷰, 트리거 등 SQL문을 수행하는 대상이 된다.
 - CREATE : 데이터베이스 생성
 - ALTER : 데이터베이스 변경
 - DROP : 데이터베이스 삭제
 - RENAME : 데이터베이스 객체이름 변경
 - TRUNCATE : 데이터베이스 저장 공간 삭제

```
-- CREATE  
CREATE TABLE DEPT01 (  
    DEPTNO NUMBER(4),  
    DNAME VARCHAR2(10),  
    LOC VARCHAR2(9)  
);
```

```
-- ALTER
ALTER TABLE DEPT01
MODIFY(DNAME VARCHAR2(30));

-- RENAME
RENAME DEPT01 TO DEPT02;

-- TRUNCATE
TRUNCATE TABLE DEPT02;

-- DROP
DROP TABLE DEPT02;
DROP TABLE DEPT02 CASCADE CONSTRAINTS; -- CASCADE CONSTRAINTS 옵션은 종속된 제약조건을 삭제한다.
```

1.3.6 DCL(Data Control Language)

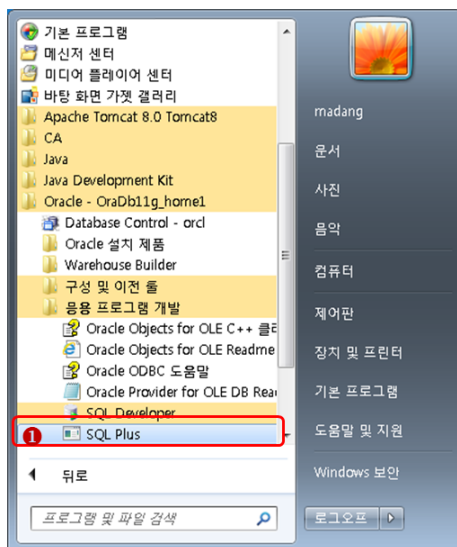
- DCL은 특정 사용자에게 권한을 부여하거나 제거하기 위해서 사용하는 명령어이다.
 - GRANT : 데이터베이스에 대한 일련의 권한 부여
 - REVOKE : 데이터베이스에 대한 일련의 권한 취소

```
GRANT CREATE SESSION TO ORAUSER01;
REVOKE CREATE SESSION FROM ORAUSER01;
```

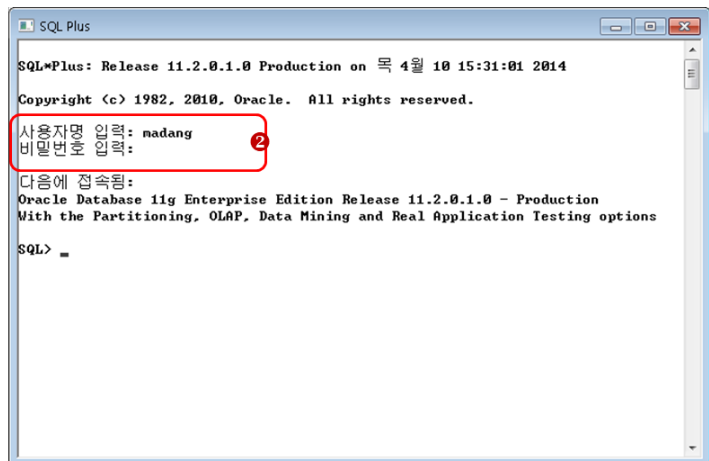
1.3.7 SQL*Plus란?

- SQL*Plus란?
 - SQL 명령문을 Command line으로 수행할 수 있는 도구이다.
 - 칼럼이나 데이터의 출력 형식을 설정하거나 환경 설정하는 기능을 제공한다.

① SQL Plus 시작



② 쿼리창 열기



1.4 SQL*Plus 로그인


1.4.1 Command 환경에서 SQL*Plus 로그인

```
-- 형식
SQLPLUS 사용자/계정암호

-- 예
SQLPLUS SCOTT/TIGER
EXIT
```

1.4.2 시스템 권한을 갖는 데이터베이스 관리자

- 데이터베이스 사용자는 오라클 계정(Account)이라는 용어와 같은 의미로 사용된다.

사용자 계정	설명
SYS	오라클 SUPER 사용자 계정이며, 데이터베이스에서 발생하는 모든 문제들을 처리할 수 있는 권한을 가지고 있다.
SYSTEM 	오라클 데이터베이스를 유지보수 및 관리할 때 사용하는 사용자 계정이다. SYS 사용자와 다른 점은 데이터베이스를 생성할 수 있는 권한이 없다.
SCOTT	처음 오라클을 사용하는 사용자의 실습을 위해 만들어 놓은 연습용 계정이다.
HR	오라클에 접근할 수 있도록 샘플로 만들어 놓은 사용자 계정이다.

1.4.3 SQL*Plus 로그인에 실패할 경우 해결 방법

(1) 사용자의 계정이 잠겨 있는 경우

```
SQLPLUS SYSTEM/MANAGER
ALTER USER SCOTT IDENTIFIED BY TIGER ACCOUNT UNLOCK;

-- CONNECT SCOTT/TIGER;
CONN SCOTT/TIGER;
```

(2) 데이터베이스가 기동되지 않는(not available) 경우

```
SQLPLUS SYS/MANAGER AS SYSDBA
STARTUP
```