

Database Programming Lab01

Prof. 심준호(jshim@sookmyung.ac.kr/새힘관407)

T.A. 이은서(les97@sookmyung.ac.kr/명신관 316B)

Team Project

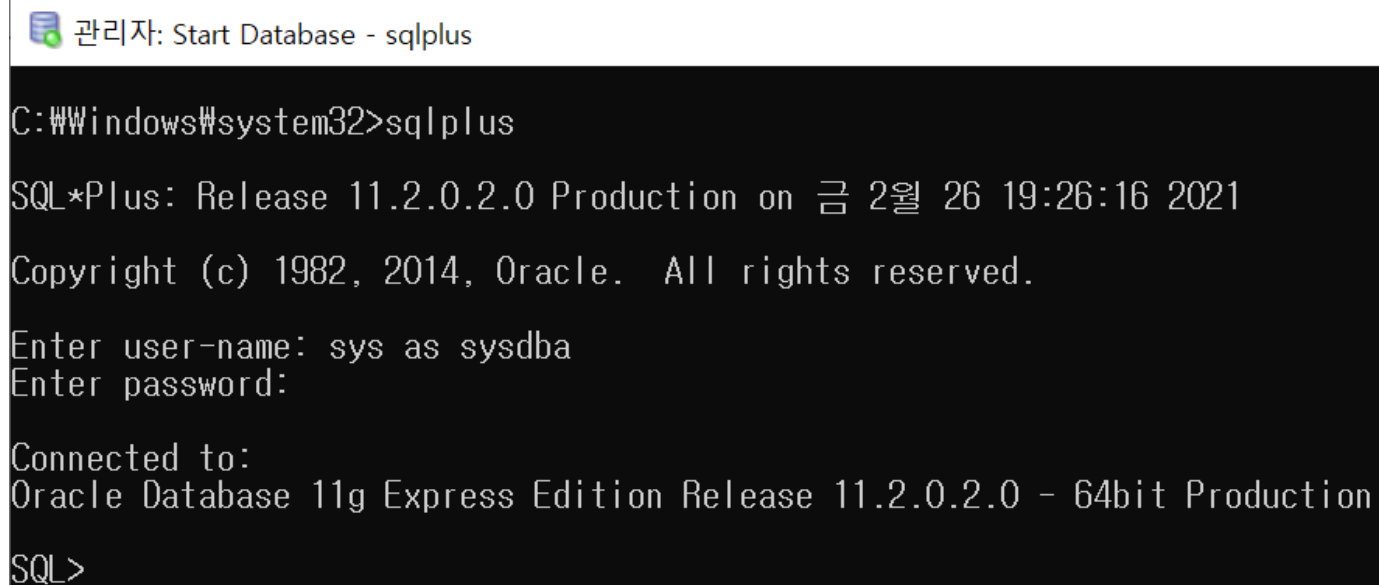
- 팀프로젝트는 중간고사 이후에 진행 예정.
- 기본은 4명이고 추후에 인원 확인 후 조정.
- 팀 구성 원하는 대로 가능.
- 팀을 못 정한 경우 따로 모아서 팀 만들어 드립니다.
- 팀프로젝트에 관한 더 자세한 안내 및 확정은 수강 인원이 확정된 이후에 안내 드릴 예정입니다.

Assignment

- DBLab01.pdf
 - 실습 및 과제 설명 자료
- Lab01.docx
 - 과제 제출 보고서
 - 제출파일명: **학번_이름.docx**
- Due
 - **다음 실습일 전날 (월요일) 11:55pm 마감**
 - Day Delay : 하루 delay 당 감점 -2
- Question
 - Snowboard 질의응답 페이지 질문 게시
 - 수요일 실습시간([1분반] 10:30~11:45 / [2분반] 1:30 ~ 2:45) 빠른 답변 가능
 - 그 외 시간도 답변 가능

Starting SQL Plus

- SQL Plus 실행
 - Start Database - sqlplus 입력
 - Window key + R -> cmd 검색 -> sqlplus 입력



```
관리자: Start Database - sqlplus

C:\Windows\system32>sqlplus

SQL*Plus: Release 11.2.0.2.0 Production on 금 2월 26 19:26:16 2021

Copyright (c) 1982, 2014, Oracle. All rights reserved.

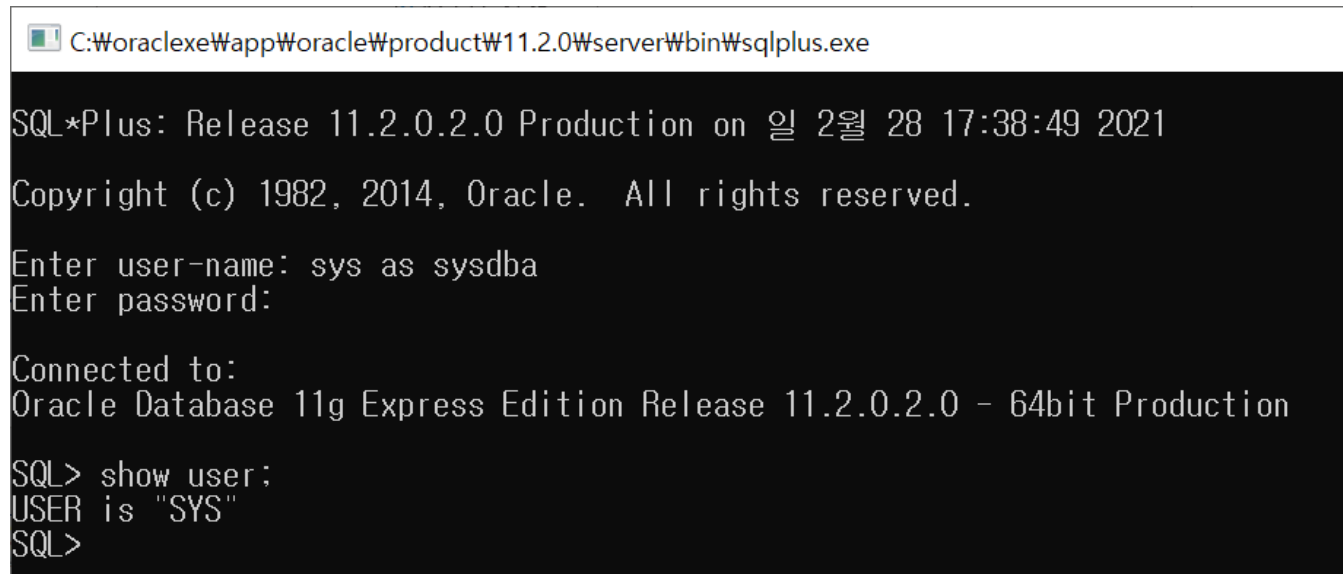
Enter user-name: sys as sysdba
Enter password:

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production

SQL>
```

Connecting as a SYSDBA

- 관리자 모드 접근
 - 사용자명 입력 : `sys as sysdba`
 - 나가기 : `exit`
 - 현재 접속한 사용자 확인
> `show user;`
 - 계정 생성, 비밀번호 변경, 권한 부여, 계정 삭제 등



```
C:\Oracle\bin\sqlplus.exe
SQL*Plus: Release 11.2.0.2.0 Production on 일 2월 28 17:38:49 2021
Copyright (c) 1982, 2014, Oracle. All rights reserved.

Enter user-name: sys as sysdba
Enter password:

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production

SQL> show user;
USER is "SYS"
SQL>
```

User Account

**실습과제1번 관련 내용

- CREATE user db identified by oracle;
 - ID : db학번 / 비밀번호 : oracle
- ALTER user db identified by ss2;
 - Db 계정의 비밀번호를 ss2로 변경
- GRANT connect, resource to db;
 - Db사용자에게 connect, resource 권한을 부여
- DROP user db [cascade];
 - Db사용자 계정을 삭제
 - Cascade 옵션은 사용자가 소유한 모든 table, integrity constraints 등을 함께 제거

```
SQL> show user;  
USER은 "SYS"입니다  
SQL> create user db identified by oracle;  
  
사용자가 생성되었습니다.  
SQL> alter user db identified by ss2;  
  
사용자가 변경되었습니다.  
SQL> grant connect, resource to db;  
  
권한이 부여되었습니다.
```

```
SQL> drop user db cascade;  
  
사용자가 삭제되었습니다.
```

Connecting as a username

- 사용자 로그인
 - 처음 로그인

```
사용자명 입력 : db  
비밀번호 입력 :
```

```
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>
```

```
사용자명 입력 : db/ss2
```

```
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>
```

- 사용자 변경해서 접속
conn id/pwd;
- 관리자모드로 변경:
conn /as sysdba;

```
SQL> show user;  
USER은 "SYS"입니다  
SQL> conn db/ss2;  
연결되었습니다.  
SQL> show user;  
USER은 "DB"입니다
```

DDL

- Data Definition Language(DDL)

데이터베이스 구조를 생성/수정/제거하는 SQL명령의 부분집합으로 CREATE TABLE, ALTER TABLE, DROP TABLE, RENAME TO 등이 있다.

- 테이블 생성

```
CREATE TABLE table_name (  
    column_name1 DATA_TYPE [Constraint],  
    column_name2 DATA_TYPE [Constraint]  
);
```

예시)

```
SQL> CREATE TABLE fruit(  
2      id NUMBER(2) NOT NULL,  
3      name VARCHAR2(20)  
4  );
```

테이블이 생성되었습니다.

Data Type

VARCHAR2(size)	문자 데이터를 동적 저장(dynamic storage)방식으로 표현함. Size는 가변길이 문자 값의 최대 길이. 최소길이 1, 최대길이 4000
CHAR(size)	문자 데이터를 정적 저장(static storage)방식으로 표현함. Size는 고정길이 문자 값의 길이. 기본길이 1, 최대길이 2000
NUMBER	38자리 정밀도 부동 소수점 숫자 표현. 기본길이 38
NUMBER(m,n)	소수점 이하 n자리, 소수점 포함 전체 m자리만 표현
DATE	날짜 , 시간 TO_CHAR(select)/TO_DATE(insert) : date 도메인과 문자 도메인을 서로 바꾸어 보여주는 포매팅 함수 INSERT INTO course_date VALUES (1, TO_DATE(yy-mm-dd,04-03-10)); DUAL table SYSDATE field : 오라클 시스템이 유지하는 현재 날짜를 저장
LONG	2G까지의 가변 길이 문자 데이터
RAW(size)	Size 길이의 번역되지 않는 byte지향 또는 binary 데이터의 저장에 이용(최대크기 2000)

SQL PLUS

- 테이블 구조 보기 : DESCRIBE[DESC] table_name;

```
SQL> DESCRIBE fruit;
```

이름	널?	유형
ID	NOT NULL	NUMBER(2)
NAME		VARCHAR2(20)

```
SQL> DESC fruit;
```

이름	널?	유형
ID	NOT NULL	NUMBER(2)
NAME		VARCHAR2(20)

- 함께 사용 가능한 모든 테이블 보기

```
SELECT * FROM tab;
```

```
SELECT * FROM tabs;
```

```
select table_name from tabs; (사용 가능한 테이블 이름만 보기)
```

```
SQL> SELECT * FROM tab;
```

TNAME	TABTYPE	CLUSTERID
FRUIT	TABLE	

SQL PLUS

- 테이블 삭제

DROP TABLE 테이블명;
purge recyclebin;(휴지통 비우기)

```
SQL> DROP TABLE fruit;  
테이블이 삭제되었습니다.
```

```
SQL> select * from tab;
```

TNAME	TABTYPE	CLUSTERID
BIN\$A4bwVsCaSNq/H+LPE5P1MA==\$0	TABLE	

```
SQL> purge recyclebin;  
휴지통이 지워졌습니다.
```

- 테이블 완전 삭제

DROP TABLE 테이블명 purge;

```
SQL> DROP TABLE fruit purge;  
테이블이 삭제되었습니다.
```

DML

- Data Manipulation Language (DML) 명령어
 - application과 DBMS 사이의 통신 수단으로 데이터 처리 연산의 집합 (데이터의 검색/삽입/삭제/변경 연산)으로 SELECT, INSERT INTO, UPDATE, DELETE FROM 등이 있다.
- 데이터 삽입 > INSERT INTO table_name [(column_name list)] VALUES (value_list)

예시1)

```
SQL> INSERT INTO fruit VALUES(1, '사과');  
1 개의 행이 만들어졌습니다.  
  
SQL> INSERT INTO fruit VALUES(2, '바나나');  
1 개의 행이 만들어졌습니다.  
  
SQL> INSERT INTO fruit VALUES(3, '자몽');  
1 개의 행이 만들어졌습니다.
```

예시2)

```
SQL> INSERT INTO fruit(id) VALUES(4);  
1 개의 행이 만들어졌습니다.  
  
SQL> INSERT INTO fruit(name) VALUES('포도');  
INSERT INTO fruit(name) VALUES('포도')  
*  
1행에 오류:  
ORA-01400: NULL을 ("DB"."FRUIT"."ID") 안에 삽입할 수 없습니다
```

SQL PLUS

- 테이블의 원하는 컬럼 보기

`SELECT column1[,column2,...] FROM table_name;`

예시) `SQL> SELECT name FROM fruit;`

NAME
사과
바나나
자몽

- 테이블의 전체 레코드 보기

`SELECT * FROM table_name;`

예시) `SQL> SELECT * FROM fruit;`

ID	NAME
1	사과
2	바나나
3	자몽
4	

SPOOL

****실습과제2번 관련 내용**

- 오라클 화면(현재 SQL에서 수행되는 모든 내용)을 파일로 저장
- spool 명령어 이후의 화면 출력을 filename.txt 파일에 저장
 - `spool C:\Users\User\Desktop\filename.txt;`
- 저장된 스크립트 파일(print.txt) SQL PLUS화면으로 출력
 - `get C:\Users\User\Desktop\print.txt;`
- 파일에 저장한 sql문 또는 pl/sql문을 실행
 - `START[또는 @] C:\Users\User\Desktop\print.txt;`
- 저장 완료
 - `spool off[;]`

Spool 예시

```
SQL> spool C:\Users\Wenfla\Desktop\dbtest\filename.txt;
SQL> get C:\Users\Wenfla\Desktop\dbtest\print.txt;
1 CREATE TABLE city(
2 id NUMBER(2),
3 name VARCHAR2(30)
4 );
5 INSERT INTO city VALUES(1, '서울');
6 INSERT INTO city VALUES(2, '뉴욕');
7 INSERT INTO city VALUES(3, '런던');
8* INSERT INTO city VALUES(4, '파리');
SQL> start C:\Users\Wenfla\Desktop\dbtest\print.txt;
```

Table created.

1 row created. SQL> select * from tab;

TNAME	TABTYPE
CLUSTERID	
CITY	TABLE

1 row created.

SQL> select * from city;

ID	NAME
1	서울
2	뉴욕
3	런던
4	파리

```
SQL> spool off;
SQL>
```

```
filename.txt - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
SQL> get C:\Users\Wenfla\Desktop\dbtest\print.txt;
1 CREATE TABLE city(
2 id NUMBER(2),
3 name VARCHAR2(30)
4 );
5 INSERT INTO city VALUES(1, '서울');
6 INSERT INTO city VALUES(2, '뉴욕');
7 INSERT INTO city VALUES(3, '런던');
8* INSERT INTO city VALUES(4, '파리');
SQL> start C:\Users\Wenfla\Desktop\dbtest\print.txt;

Table created.

1 row created.

1 row created.

1 row created.

SQL> select * from tab;

TNAME                                TABTYPE
-----
CLUSTERID
CITY                                  TABLE

SQL> select * from city;

ID NAME
-----
1 서울
2 뉴욕
3 런던
4 파리

SQL> spool off;
```

실습 1 & 2

[실습1] 계정 로그인 과정을 캡처하세요.(SQL문 + 결과문)

[과정]

- (1) 'db학번' 계정 생성
- (2) Connect, resource 권한 부여
- (3) 해당 계정 로그인

[실습2] 시나리오대로 실행한 내용을 스크랩하고, 스크랩 파일 내용을 보고서에 그대로 붙이세요.

** 스크랩파일(학번.txt)를 아래에 붙일 때, 스크랩 순서 명령어에 해당하는 sql문을 굵게 변경하세요.(SQL> ~)

(안할 시 감점 -2)

실습 2 - 시나리오(1/2)

1) 5개 테이블 생성과 각 테이블 당 5개 이상 레코드 삽입하는 input.txt 작성

Table	Columns	Records
STUDENT	<u>학번</u> (ex: s_id), 이름, 학년, 대학, 전공	(200101, '김가나', 1, '숙명', '컴과') (200102, '이다라', 2, '숙명', '소용') (200103, '박마바', 2, '숙명', '컴과') (200104, '최사아', 3, '숙명', '컴과') (200105, '정자차', 4, '숙명', '소용')
PROFESSOR	<u>교번</u> (ex: p_id), 이름, 대학, 전공	(201, '심준호', '숙명', '컴과') (202, '박영훈', '숙명', '컴과') (203, '최종원', '숙명', '컴과') (204, '김윤진', '숙명', '소용') (205, '박숙영', '숙명', '소용') (206, '채희준', '숙명', '소용')
COURSE	<u>과목번호</u> , 과목이름	(200011, '데이터베이스') (200012, '시스템종합설계') (200013, '인공지능') (200014, '컴파일러') (200015, '컴퓨터네트워크2') (200016, '컴퓨터특강')
ENROLL	<u>학번</u> , <u>과목번호</u> - 수강신청 정보 테이블 (*student, course 관련 있음)	(200101, 200011) (200101, 200013) (200102, 200013) (200102, 200014) (200103, 200014) (200103, 200011)
TEACH	<u>교번</u> , <u>과목번호</u> - 교수의 강의 정보 테이블 (*professor, course 관련 있음)	(201, 200012) (201, 200014) (202, 200011) (202, 200015) (203, 200013)

(* 학번, 교번, 과목번호 : 제약조건으로 NOT NULL!!)

실습 2 - 시나리오(2/2)

- 2) SQL PLUS에서 스푼 시작(파일명 : 학번.txt)
- 3) 현재 사용자명 보여주기
- 4) 현재 사용 가능한 테이블 리스트 보여주기
- 5) input.txt 파일을 화면으로 출력
- 6) 스크립트 파일(input.txt) 실행
- 7) 현재 사용 가능한 테이블 리스트 보여주기
- 8) course 테이블 구조 보여주기
- 9) student 테이블에 저장된 데이터의 학번, 전공 보여주기
- 10) 스푼 완료

Submission

- Lab01.docx
 - 제출파일명: **학번_이름.docx**
- Due
 - **2021.03.22(월) 11:55pm전 까지**
 - Day Delay : 하루 delay 당 감점 -2