Oracle Installation

CSE 4308
DATABASE MANAGEMENT SYSTEMS LAB

AUGUST 8, 2023

1   Introduction

SQL stands for *Structured Query Language*. It is used to *manage data* stored in a *Relational Database Management System*. It is useful in handling *structured data*.

2   Getting Started with Oracle 11g XE

Throughout the course, we will use Oracle 11g XE database. Oracle Database Express Edition is a free, smaller-footprint edition of Oracle Database. It is easy to install and manage. Using the tools provided in Oracle Database XE, you can:

- Administer the database

- Create tables, views, and other database objects

- Import, export, and view table data

- Run queries and SQL scripts

Whenever you are installing it on your own PC, make sure to remember the password. To open the SQL command line, open Command Prompt and type `sqlplus`, or Search 'run SQL command line' in the search bar.
SQLPlus is the most basic Oracle Database utility, with a basic command-line interface, commonly used by users, administrators, and programmers. Opening the SQL command line will ask you to enter the username and password. The default username for login is 'system' and the password is the one that you provided during installation. Remember that, when writing the password, the characters will not be shown as Oracle keeps even the number of characters of the password hidden so that anyone else cannot peek over your shoulder and see it.

3   Creating a New User

To perform database operations, you have to create a user. To create a new user, we need to follow a sequence of operations:

- Log in as `system`

- Create a new user with a password

- Grant sufficient permissions/privileges

The major types of privileges among some others are:

- `CREATE SESSION` → Allows user to log in

- `RESOURCE` → Allows user to manipulate tables

- `TABLESPACE` → Ensures that a user has disk space allocated in the system to create and modify tables and data

- `DBA` → Database administrator privilege

The following template statement can be used to create a new user:

```
CREATE USER username IDENTIFIED BY password;
```

Here, `username` and `password` should be replaced by your intended username and password. For example, if you want to create a user 'dbms_170042001' with password 'cse4308', the command would be:

```
CREATE USER dbms_170042001 IDENTIFIED BY cse4308;
```

Again, the following template statement can be used to grant different privileges to a user:

```
GRANT privilege1, privilege2, ... TO username;
```

Although not recommended in production, we can use the following query to grant all privileges to the user:

```
GRANT ALL PRIVILEGES TO username;
```

This query will allow the user to perform any operations on the database. For example, if we want to grant all privileges to 'dbms_170042001', the command would be:

```
GRANT ALL PRIVILEGES TO dbms_170042001
```

After creating the user and granting them privileges, you can connect to the database using the CONNECT command. The template for the command is:

```
CONNECT username/password;
```

Here, `username` should be replaced by the username of the account and `password` should be replaced by the password for that account. For example, to log in using 'dbms_170042001' would be:

```
CONNECT dbms_170042001/cse4308;
```

Remember that, the semicolon denotes the end of a statement. It is not a part of the password or any other statement.
Alternatively, you can write:

```
CONNECT
```

Then you will be prompted to provide a username and the corresponding password to log in.

4 Testing with a Table

To create a table, you have to write the following query:

```
CREATE TABLE EXAMPLE
(
  NATIONAL_ID NUMBER NOT NULL,
  NAME VARCHAR2(50) NOT NULL,
  BIRTH_DATE DATE
);
```

We can type the following command to insert new records into our 'EXAMPLE' table:

```
INSERT INTO EXAMPLE VALUES (2015001, 'W', '01-JAN-81');
INSERT INTO EXAMPLE VALUES (2015002, 'X', '04-APR-83');
INSERT INTO EXAMPLE VALUES (2015003, 'Y', '14-JUN-82');
INSERT INTO EXAMPLE VALUES (2015004, 'Z', '27-SEP-81');
```

To find all records and their information:

```
SELECT * FROM EXAMPLE;
```

5   Tips and Tricks

- To check which user is currently logged in, write the following command:

```
SHOW USER;
```

- If you face problem connecting with the 'system', then try this command:

```
CONNECT SYS/SYS AS SYSDBA;
```

- In case you have a huge pile of text in your console and you want to clean things up, write the following:

```
CL SCR;
```