Logistics in Project 1

Command history

Sqlplus is missing command history. That means if you want to repeat a previous command, you have to retype it from scratch. Over time, you will find that it is annoying. Here are two ways to get command history:

```
% rlwrap sqlplus
```

```
% gqlplus
```

Use either rlwrap or gqlplus, and you should have command history.

Properly Quitting sqlplus

It is important to quit sqlplus properly. Otherwise, you may find that you get an error that there are too many active sessions and sqlplus will not allow you to login. Here is the command to quit:

```
SQL> QUIT
```

Always type the above to terminate a session. You should see a message like the following:

```
SQL> Disconnected from Oracle Database 11g Enterprise Edition Release
11.2.0.2.0 - 64bit Production
```

If you do get an error about too many active sessions when logging into Oracle, try doing the following:

```
% ps -ux
```

Look for processes that appear to be sqlplus related. Note their process ID (a number in the 2nd column). Use the kill command to kill them:

```
% kill pid
```

In the above pid is the process ID of the process you want to kill.

If the above does not resolve the issue, look for all windows you used for logging to CAEN. Do the above on all of them.

If the above still does not resolve the issue, you may need to contact ITD Help or CAEN Hotline and request them to terminate your Oracle sessions.

Reading SQL commands from External Files

Use the START command within sqlplus

```
SQL> START filename
```

You may find it useful to set the following option before doing the above:

```
SQL> SET ECHO ON
```

That way, sqlplus will echo on the terminal every command that is read from an external file. If there is an error, you will be able to better tell the location of the error.

Getting a list of all the Tables in Oracle

You can use the following command to retrieve the list of all the tables that are accessible to you. Some will belong to other users. The owner information will be shown.

```
SELECT owner, table_name FROM all_tables;
```

Dropping Tables, Triggers, Views

User the DROP command. E.g.,

```
DROP TABLE tablename;

DROP TRIGGER triggername;

DROP VIEW viewname;
```

It is useful to put the DROP commands in a batch file to clean up the database you are working on. That way, you can re-create the database easily without getting errors as you are experimenting with the database design.

Normally, DROP commands must be executed in reverse order to CREATE commands to avoid violating constraints. Also see later discussion on CASCADE option to DROP TABLE command in the discussion on circular dependencies. That can be really useful to get around the problem of ordering DROP commands properly.

Finding Schema of a Table

You can find the schema of a table by using the SQLPLus DESC command.

```
DESC <TableName>
```

For example, if you have a table named userinfo with five columns, here is what you may see:

```
SQL> DESC gsiid.PUBLIC_USER_INFORMATION
```

The above displays the schema of the table PUBLIC_USER_INFORMATION belonging to user gsiid. If it is your own table, you can omit the userid.

```
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USER_ID                                            VARCHAR2(100)
 FIRST_NAME                                         VARCHAR2(100)
 LAST_NAME                                          VARCHAR2(100)
 YEAR_OF_BIRTH                                      NUMBER(38)
 MONTH_OF_BIRTH                                     NUMBER(38)
```

Viewing Very Wide and Large Tables in Excel

If you try to run a select query on PUBLIC_USER_INFORMATION in sqlplus, you will find that it is very hard to understand the data in the terminal. It would seem that it is not in columns, with all the values printing on lines by themselves.

The problem is that most of the columns such as FIRST_NAME or LAST_NAME in PUBLIC_USER_INFORMATON are defined to be long strings, for example, 100 characters wide. Sqlplus assumes that lines are 80 characters wide. It will insert newlines in the output after 80 characters.

To get around that, you can issue the following commands:

```
SQL> set line 10000
```

This sets line width to a large value. Now, you should be able to see an entire row on one line. Of course, it will wrap around on your terminal since your terminal is not that wide. But, it will be easier to tell rows apart.

You can also change the column separator. For example:

SQL> set colsep '|'

The above makes the column separator a |.

Of course, what you would really like to do is view the contents of a table in Excel. To do that, you can do something like the following:

```
set colsep ','     -- separate columns with a comma
set pagesize 0   -- No header rows
set trimspool on -- remove trailing blanks
set headsep off  -- this may or may not be useful...depends on your headings.
set linesize 10000   -- linesize should be > sum of the column widths
spool userinfo.csv    -- output will be spooled to this file
SELECT * FROM TABLENAME;         -- type your query here
spool off
```

In the above, replace TABLENAME with the name of the table that you want to spool out to a CSV file. The output from the SELECT command will be spooled to userinfo.csv, or whatever file you specify.

In the above, we set the column separator to a comma (CSV stands for comma-separated values). We set a few more options in sqlplus to produce a better output. The most important is to have a sufficiently long linesize so that lines are not chopped up. At the end, we set spool to off so that spooling stops.

You should be able to import or open a CSV file into Excel , LibreOffice, or Google Docs spreadsheet.

Circular Dependencies in Foreign Keys

You may have a situation where TABLE A contains a foreign key to TABLE B and TABLE B contains a foreign key to TABLE A.

The difficulty you will run into in that case is that you will not be able to create the tables. If you attempt to create TABLE A, sqlplus will complain that there is no TABLE B. If you create TABLE B, it will complain that there is no TABLE A.

The solution is to create the tables without the foreign key constraint Then, you should be able to create table A and B. Then, use the ALTER TABLE command to add the foreign key constraint to table A and B. Here is an example:

What we want to get:

```
CREATE TABLE A (id: INTEGER PRIMARY KEY, bid: INTEGER REFERENCES B);

CREATE TABLE B(bid: INTEGER PRIMARY KEY, aid: INTEGER REFERENCES A);
```

But the above do not work because of circular dependency. The first command will fail.

Solution:

```
CREATE TABLE A (id INTEGER PRIMARY KEY, bid INTEGER);

CREATE TABLE B(bid INTEGER PRIMARY KEY, aid INTEGER);
```

The above creates table A without a dependency on table B.

Now, add in the dependency:

```
ALTER TABLE A ADD CONSTRAINT ARefsB FOREIGN KEY (bid) REFERENCES B
INITIALLY DEFERRED DEFERRABLE;

ALTER TABLE B ADD CONSTRAINT ARefsB FOREIGN KEY (aid) REFERENCES A
INITIALLY DEFERRED DEFERRABLE;
```

The INITIALLY DEFERRED DEFERRABLE is important. It says that constraint checking should only be done at COMMIT points in a transaction, not within a transaction. Without that, you will not be able to add records to either table A or B.

To add records to table A and B, you can do the following:

```
SET AUTOCOMMIT OFF

INSERT INTO A VALUES (1, 2);

INSERT INTO B VALUES (2, 1);

COMMIT

SET AUTOCOMMIT ON
```

The above would add an entity with aid 1 to table A with a reference to entity 2 in table B. Table B's entity 2 points to entity 1 in table 1. Without deferred constraint checking, the first INSERT would have failed.

The SET AUTOCOMMIT commands turn off or on auto-commit mode. When AUTOCOMMIT is ON (default), each SQL command is a transaction by itself. Constraints are always checked at the end of a transaction. When AUTOCOMMIT is OFF, SQL commands are not committed to the database until a COMMIT command is seen. Deferred constraints are checked only at COMMIT, rather than on each insert.

Normally, you want AUTOCOMMIT ON when interacting with sqlplus interactively. But, it can be helpful to turn it off temporarily to insert multiple records as a batch and do constraint checking only at the end of the entire batch.

To drop the above tables, you will run into problems. Sqlplus will not allow you to drop either table because other table refers to it. The solution is the following:

```
DROP TABLE A CASCADE CONSTRAINTS;

DROP TABLE B CASCADE CONSTRAINTS;
```

ON DELETE NO ACTION

In Oracle, there is no ON DELETE NO ACTION. It is not needed because that is the default. Remember that ON DELETE NO ACTION means that if there is going to be a foreign key constraint violation, do not allow the transaction to execute (no action). The transaction is aborted.

If you include ON DELETE NO ACTION clause in a CREATE TABLE command, sqlplus will report an error.

If you want to cascade the change, you can specify ON DELETE CASCADE. That is acceptable in Oracle.

Enumerated Type-Style constraints

Project 1 specs at the time of this writing state that ALBUM_VISIBILITY attribute for an ALBUM should be one of several given values. Unfortunately, the specs may have one of the attribute values as incorrect. ONLY_ME should be MYSELF. Use VARCHAR(size) for the type and add a CHECK constraint that it is one of the given values as follows:

```
TABLE ALBUMS (

….
CONSTRAINT constraintname CHECK ((ALBUM_VISIBILITY = 'MYSELF') OR … )

)
```

The size value in VARCHAR needs to be only large enough to hold the possible values.

Connecting to Oracle from a non-CAEN computer

(Not supported - but provided for the adventurous)

Normally, we assume that you are logging in from a CAEN machine directly. This is unsupported (do not ask us for help), but it may be possible to connect remotely. Below are the instructions for Mac OS X, Mountain Lion that worked for getting sqlplus working.

1. Download the 32-bit versions of instantclient-basic and instantclient-sqlplus from Oracle's web site. You may have to create an account (free) with Oracle. At the time of this writing, the files were:

instantclient-basic-10.2.0.4.0-macosx-x86.zip

instantclient-sqlplus-10.2.0.4.0-macosx-x86.zip

Unzip both of them in a place of your choice. Note the location of the directory in which they are unzipped. Add that directory to your LD_LIBRARY_PATH environment and to your PATH environment variable by modifying the .bash_profile file in your home directory.

Once you do that, you should be able to execute sqlplus.

But, sqlplus is just a client to the database. It needs to connect to a database for you to do anything. Here is the command to connect to your course database:

```
sqlplus userid/password@forktail.dsc.umich.edu/COURSEDB
```

OR

```
sqlplus userid@forktail.dsc.umich.edu/COURSEDB
```

(The latter did not work for me, but is theoretically supposed to work. Let me know if you find a way to not have to type the password in the command)

You can also install rlwrap on the Mac as follows. First install macports (port). Then, do the following:

```
% sudo port install rlwrap
```

Then, the following should work:

```
% rlwrap sqlplus userid/password@forktail.dsc.umich.edu/COURSEDB
```

Similar steps may work for Windows. Also the @.forktail.dsc.umich.edu/COURSEDB is the string that identifies the database server and database we are using. If you try out another client, such as Oracle Developer, you may need it.