COMP2714 - RELATIONAL DATABASE SYSTEMS
Assignment 03 – SQL DML Subqueries, Aggregates, Views
Due at START of your Lab in week Oct 03 (Tue Oct 04) - Oct 07 (Fri), 2016

You are to work in groups of two. Assignments must be done using Oracle SQL*Plus. Make sure you save all your work in sql script files, and keep backup copies of these scripts!! [Refer to Lab01_sample.sql as illustrative example.]

Assignment submissions must be clearly labeled with identification details, printed black ink on white paper, stapled and submitted to the designated locations; otherwise marks will be deducted. See submission requirement details below.

Coding guideline for ALL assignments: [e.g. Lab01_sample.sql]_ Otherwise10% of the total marks will be deducted.
- UPPER CASE for all SQL reserved words, and Mixed or Lower Cases for names you defined.
- Start each SQL clause on a new line; indent subsequent additional lines of the same SQL clause; avoid use of long lines but appropriately break into multiple lines with indentation; use white space to align code to improve readability.
- Use comments to label your questions, add notes and explanations where appropriate.
- Use single quotes for character strings and date values, no quotes for numeric values, NULL keyword instead of ''.

Some useful tips:
- Remember to SET ECHO ON in your SQL*Plus environment, so that your script file content will be included in your spool output file. Make sure SPOOL OFF is included near the end of your script file so you will not loose output.
- Check out (on the Internet) some Oracle built-in functions such as TO_DATE, TO_CHAR, TO_NUMBER, DUAL, String Concatenation, etc. as these may come in handy.

You have to submit BOTH a printed copy and a soft (electronic) copy of the SPOOL output file, with file extension .txt (unless otherwise determined by your lab instructor); if either copy is omitted, assignment is deemed NOT submitted:
- **For BOTH copies, label each answer clearly with the proper question number; otherwise, 10% of the total marks will be deducted. For queries, include each query result output immediately after each query answer.**
- **For the printed copy, NO cover page required; include Lastname(s), Firstname(s), student number(s), course number, assignment number and set letter at the beginning as comments (See Lab01 sample). The pages should be STAPLED together, printed with black ink on white paper [no color or grey]. Otherwise, 10% of total marks will be deducted.**
- **Print with FIXED-size font type like Courier New, with font size 11 or greater. To save printing paper, and if a result output is too long, you can just keep the first 2 rows and delete the rest of that result set output before printing.**
- **For the soft copy spool file, submit it to sharein I:\COMP\2714\Day\2<set>\Asn3 where 2<set> is the subfolder for your set. Make sure you submit to the proper subfolder for your set; otherwise, it will be treated as NOT SUBMITTED. Your spool file should be named Asn3_<LastName><FirstInitial>.txt. Include both partners' names as appropriate.**

Given the following database, use a **single** SQL statement to provide the information required by each query question 1 to 10:
(**Note**: you can run the script Asn3SetupHotels.sql to create the database tables with some sample data; **NO need to print this!!**)

Hotel (hotelNo, hotelName, hotelAddress)
Room (roomNo, hotelNo, type, price)
Guest (guestNo, guestName, guestAddress)
Booking (hotelNo, guestNo, dateFrom, dateTo, roomNo)

Queries Connolly & Begg 6/e, p.182: (**If you have questions, ask your instructors early. Do not over-spend your time.**
& p.220 **Practice a systematic problem solving approach as demonstrated in the lectures.**)

For each question, start with the question statement in the textbook, then apply additional requirements given into a single question.

1. (6.13 - 5 mks) What is the average price of a room in London? .a) Do this using JOIN, b) Do this using IN subquery.
2. (6.15 - 5 mks) Use the month of 2016-10 instead of August, and do the query for each hotel, listing in hotelName order: i.e. How many different guests have made bookings for 2016-10 for each hotel?
3. (6.18 - 5 mks) Use 2016-10-06 as today's date. Include all 'Grosvenor' hotels. List in hotelNo, roomNo order.
4. (6.20 - 5 mks) Use 2016-10-06 as today's date. Include all 'Grosvenor' hotels. List in hotelNo, roomNo order. Use NOT IN.
5. (6.20 - 5 mks) Use 2016-10-06 as today's date. Include all 'Grosvenor' hotels. List in hotelNo, roomNo order. Use NOT EXISTS.
6. (6.20 - 5 mks) Use 2016-10-06 as today's date. Include all 'Grosvenor' hotels. List in hotelNo, roomNo order. Use LEFT JOIN.
7. (6.20 - 5 mks) Use 2016-10-06 as today's date. Include all 'Grosvenor' hotels. List in hotelNo, roomNo order. Use MINUS.
8. (6.24 - 5 mks) Use the month of 2016-10 instead of August.
9. (6.26 - 5 mks) Use 2016-10-06 as today's date. List hotelNo, hotelName and LostIncome, in hotelNo order.
10. (7.14 - 5 mks, p.220) Use 2016-10-06 as today, and guestAccount as view name. [The account info for each guest means: Room (roomNo), Name (guestName), Check-in (dateFrom), Check-out (dateTo=today), Rate (price), #Days (calculate from dateTo and dateFrom), and Total (calculate from dateTo, dateFrom and price).] Add a query to test the view.

0. (7.15 and 7.16 - **Optional** practice question on DCL data control language; no assigned marks)

NOTE: Expression to determine if a Booking row contains the current date (e.g. 2016-10-06):

```
dateFrom <= DATE'2016-10-06' AND (dateTo IS NULL OR dateTo >= DATE'2016-10-06')
```

Expression to determine if a Booking row contains one or more days of a particular month (e.g. 2016-10):

```
(dateTo IS NULL OR dateTo >= DATE'2016-10-01') AND dateFrom < DATE'2016-11-01'
```

Total: 50 marks