

COSC 344

Lab for Week 4

Overview

The purpose of this lab and the next two is to get you familiar with Oracle's SQL commands.

Hint:

You may want to have an editor window (e.g. emacs or vim) up and enter the commands into a file. Then you can save the file and load it as a script. This can save a lot of typing. Make sure you save it after each alteration. You can save each query statement in a separated file named "q##.sql" where ## is the ID of the query (e.g. q01.sql). It will be handy for you to prepare the lab assessment.

When you retrieve data from a table, you may find that the output of the retrieved data is untidy as one row in the table is broken into multiple lines in the terminal. This is due to the constraint on line size in DBMS Output. You can use the following command at the SQL> prompt to change the line size:

```
SQL> set linesize 1000; // this changes the line size to 1000 characters.
```

Date and Time

Create a table with a column for a date and a column for a date and time.

Populate it with about 4 dates and date/times using the TO_DATE function. You may want to review slides 21 and 22 of Lecture 5 for the DATE data type and the TO_DATE and TO_CHAR functions.

Experiment with TO_CHAR to output the date in various formats using a SELECT statement. For example, output your dates as 31-01-2000, 31-Jan-2000, 31/01/2000, 01/31/2000 (US style). List some times in various formats such as 10:30:45, 10:30. Also list some dates and times together. Refer to the lecture notes for the formatting.

Loading the Tables Used in Lectures

On a lab computer, open a terminal. On your own computer, you need to log into titanium first. Create a sub-directory named Lab4, and then change to this subdirectory. Copy the files, *company.sql* and *order.sql*, from /coursework/344/pickup/oracle-sql to your current directory Lab4 using the following commands **at the Linux prompt**

```
cp /coursework/344/pickup/oracle-sql/company.sql .
```

```
cp /coursework/344/pickup/oracle-sql/order.sql .
```

These two files contain SQL commands to drop tables, create tables and populate tables. They are the same as the tables used in lectures. Connect to Oracle from the directory where these two .sql files are placed, and then execute the scripts at the SQL> prompt using @company.sql and @order.sql to create and populate the tables for the next part of the lab.

Take a few minutes to study the *company.sql* script.

In particular, look at the definition of the EMPLOYEE table. Do you understand the various pieces? Why were some of the constraints named? Why was *superssn_cnst* disabled until after data load?

Executing Queries

Below is a set of query statements. Work out the SQL queries to get the desired results. The correct results are shown in the section entitled “Query Results” at the end of this document. Note that the order of the rows in the output you produced may be different from the order in the answer. Don’t worry about the order as long as you get the correct set of data. If you get stuck, ask for help.

A copy of the COMPANY and ORDER databases is available at
https://www.cs.otago.ac.nz/cosc344/resources/company_db.pdf
https://www.cs.otago.ac.nz/cosc344/resources/sales_db.pdf

It is very helpful to check the tables in these two databases when working on the SQL queries, especially for the query to retrieve data from multiple tables.

1. Use the DESCRIBE command to get the attributes and data types of the *employee* table.
2. List all the daughters and spouses in the *dependent* table. Can you think of a second way to express this query. Would it be appropriate to use DISTINCT for the dependent name?
3. List the projects that do not have a project location.
4. List the name of the manager for each department and the name of the department they manage.

5. List the names of employees who worked on a project less than 18 hours.
6. List the names of employees whose last name begins with 'W'.
7. Use 'IN' to list the names of salespeople located in San Jose or Barcelona.
8. Use 'BETWEEN' to list the customers who placed orders with amounts between 1500 and 5000.
9. How many rows does the orders table have?
10. What is the average amount of all the orders?

Query Results

2. Assuming you only project the dependent_name, the result is:

Alice
Joy
Abner
Alice
Elizabeth

3. Assuming you only project pname, the result is:

NonProject

4. Franklin Wong Research
Jennifer Wallace Administration
James Borg Headquarters

5. Alicia Zelaya
Franklin Wong
James Borg
Jennifer Wallace
John Smith

6. Franklin Wong
Jennifer Wallace

7. Serres
Rifkin

8. Pereira 1900.1
Giovanni 1713.23
Clemens 4723

9. There are 10 rows

10. The average is 2665.84

Assessment: 18 marks, due at 5pm March 30

Query 2 to Query 10 will be assessed.

If you saved the statement for each query in a separated file named “q##.sql”, you can use the following command from the **Linux prompt** to concatenate the 9 queries into one file:

```
cat q???.sql > Lab4.sql
```

Record an Oracle session using SQL spool with the following commands at the SQL> prompt. (Refer to the lab in Week 2 on how to use spool).

```
SQL> spool Lab4;  
SQL> @Lab4.sql  
SQL> spool off;
```

A file named Lab4.lst will be created. Submit Lab4.sql and Lab4.lst in Blackboard.