L2-W2-DBS301-select-range-order

**Due Friday midnight of week 2**

*Name: Avinash Singh*

*ID: 115408163*

*Course: DBS301*

*Date: 15-09-2017*

*LAB 2*

*STEP 1: rename the file to L2-your id name*

*STEP 2: Put the SQL and the results after each question below*

*STEP 3: Submit on Blackboard.*

1 Display the employee\_id, last name and salary of employees earning in the range of $8000 to $15,000. Sort the output by top salaries first and then by last name.

**Query**: **SELECT employee\_id AS "Emp ID",**

**last\_name AS "Last Name",**

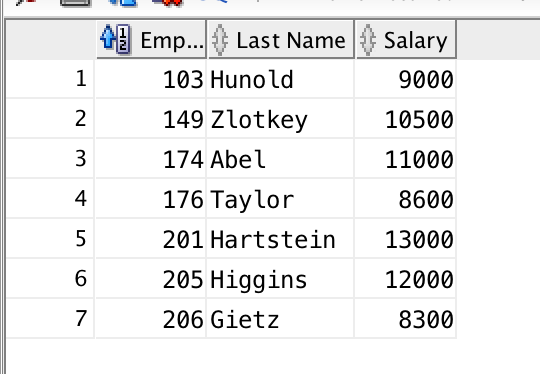
**salary AS "Salary"**

**FROM employees**

**WHERE salary >= 8000 AND salary <= 15000**

**ORDER BY salary DESC, last\_name;**

**Result:**

****

2 Modify previous query (#1) so that additional condition is to display only if they work as Programmers or Sales Representatives. Use same sorting as before.

**Query**: **SELECT employee\_id AS "Emp ID",**

**last\_name AS "Last Name",**

**salary AS "Salary",**

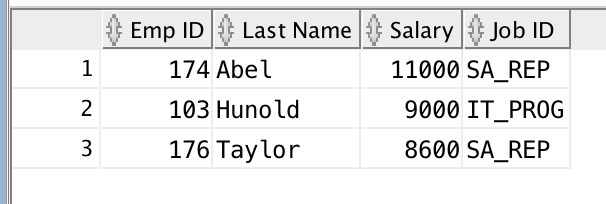
**job\_id AS "Job ID"**

**FROM employees**

**WHERE (salary >= 8000 AND salary <= 15000)**

**AND (job\_id LIKE 'SA\_REP' OR job\_ID LIKE 'IT\_PROG')**

**ORDER BY salary DESC, last\_name;**

**Result: **

3 The Human Resources department wants to find high salary and low salary employees. Modify previous query (#2) so that it displays the same job titles but for people who earn outside the given salary range from question 1. Use same sorting as before.

**Query**: **SELECT employee\_id AS "Emp ID",**

**last\_name AS "Last Name",**

**salary AS "Salary",**

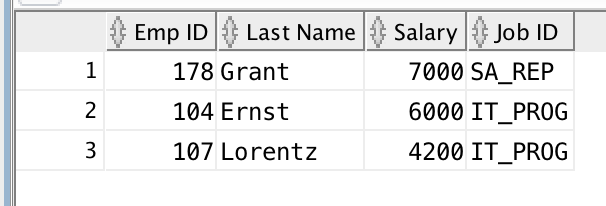
**job\_id AS "Job ID"**

**FROM employees**

**WHERE (salary < 8000 OR salary > 15000)**

**AND (job\_id LIKE 'SA\_REP' OR job\_ID LIKE 'IT\_PROG')**

**ORDER BY salary DESC, last\_name;**

**Result: **

4 The company needs a list of long term employees, in order to give them a thank you dinner. Display the last name, job\_id and salary of employees hired before 1998. List the most recently hired employees first.

**Query**: **SELECT last\_name AS "Last Name",**

**salary AS "Salary",**

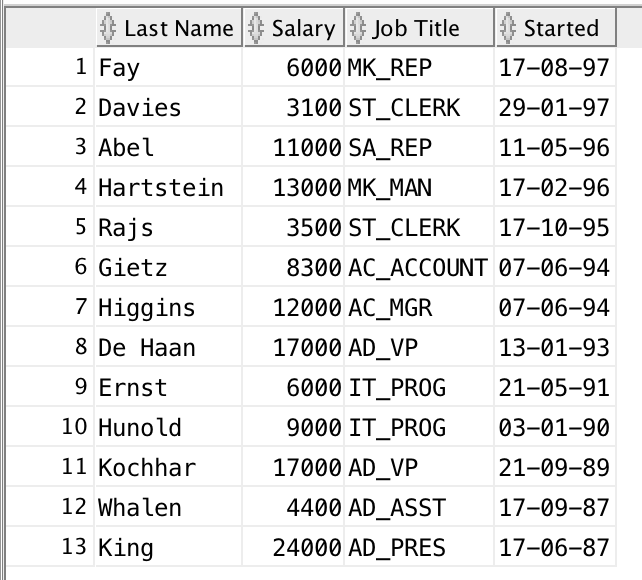
**job\_id AS "Job Title",**

**hire\_date AS "Started"**

**FROM employees**

**WHERE (hire\_date < DATE '1998-01-01')**

**ORDER BY hire\_date DESC;**

**Result: **

5 Modify previous query (#4) so that it displays only employees earning more than $10,000. List the output by job title alphabetically and then by highest paid employees.

**Query**:

**SELECT last\_name AS "Last Name",**

**salary AS "Salary",**

**job\_id AS "Job Title",**

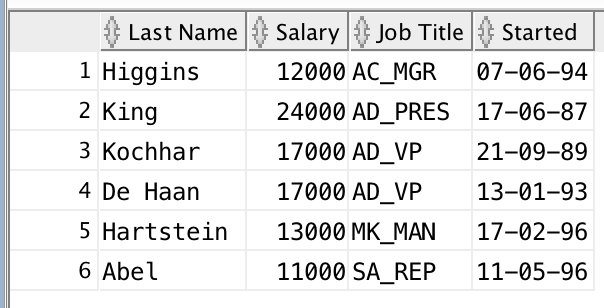
**hire\_date AS "Started"**

**FROM employees**

**WHERE (hire\_date < DATE '1998-01-01')**

**AND salary > 10000.0**

**ORDER BY Job\_ID, salary DESC;**

**Result: **

6 Display the job titles and full names of employees whose first name contains an ‘e’ or ‘E’ anywhere. The output should look like:

Job Title Full name

--------------------------------------

AD\_VP Neena Kochhar

… more rows

**Query**:

**SELECT Job\_id AS "Job Title",**

**First\_Name || ' ' || Last\_Name AS "Full Name"**

**FROM employees**

**WHERE First\_Name LIKE '%e%'**

**OR first\_name LIKE '%E%';**

**Result: **

7 Create a report to display last name, salary, and commission percent for all employees that earn a commission.

**Query**:

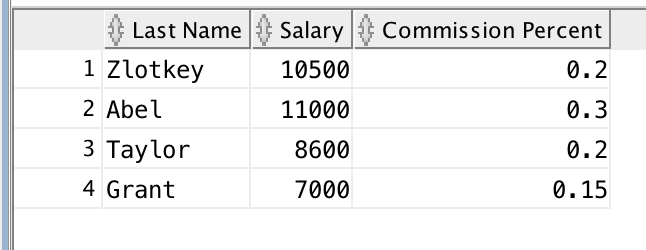
**SELECT last\_name AS "Last Name",**

**salary AS "Salary",**

**commission\_pct AS "Commission Percent"**

**FROM employees**

**WHERE commission\_pct IS NOT NULL;**

**Result: **

8 Do the same as question 7, but put the report in order of descending salaries.

**Query**:

**SELECT last\_name AS "Last Name",**

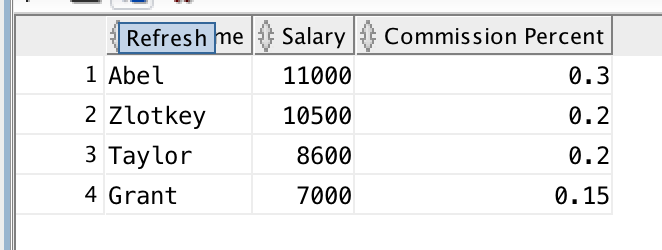
**salary AS "Salary",**

**commission\_pct AS "Commission Percent"**

**FROM employees**

**WHERE commission\_pct IS NOT NULL**

**ORDER BY salary DESC;**

**Result: **

9 Do the same as 8, but use a numeric value instead of a column name to do the sorting.

**Query**:

**SELECT last\_name AS "Last Name",**

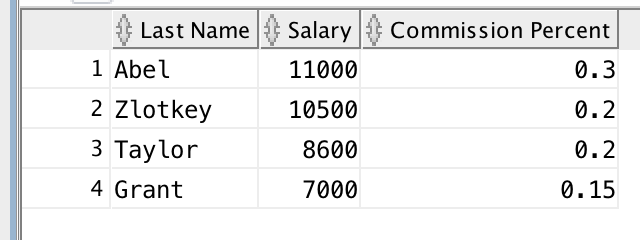
**salary AS "Salary",**

**commission\_pct AS "Commission Percent"**

**FROM employees**

**WHERE commission\_pct IS NOT NULL**

**ORDER BY 2 DESC**;

**Result: **