**Lab 3**

Clear screenshots of successful run of SQL query and output is required in a single file. Zero will be assigned otherwise. You may use SQL developer or SQL plus. If you want to be independent of mySeneca apps or Seneca Oracle instance, install Oracle XE in your laptop and use SQL plus.

Please remember to upload the correct file before submitting - you only get one chance to submit. The Lab 3 submission link will not be available after deadline in the Assignments section.

**Allow enough time to upload or deal with unexpected issues.** **Do not wait for the last moment** since there are transmission time/queuing delay/processing time etc. from your machine to the blackboard server.

**Late penalty is 100%.**

1. Display the difference between the Average pay and Lowest pay in the company among employees.

Name this result *Real Amount*.

**SELECT AVG(salary) - MIN(salary) AS "Real Amount" FROM employees;**

A screenshot of a computer

Description automatically generated

1. Display the department number and Highest, Lowest and Average pay per each department. Name these results *High, Low* and *Avg.*

Sort the output so that department with highest average salary are shown first.

**SELECT department\_id, MAX(salary) AS High, MIN(salary) AS Low, AVG(salary) AS Avg FROM employees GROUP BY department\_id ORDER BY AVG(salary) DESC;**

*A screenshot of a computer

Description automatically generated*

1. Display how many people work the same job in the same department. Name these results *Dept#, Job* and *HowMany.* Include only jobs that involve more than one person.

Sort the output so that jobs with the most people involved are shown first.

**SELECT department\_id AS Dept#, job\_id AS Job, COUNT(\*) AS "How many" FROM employees GROUP BY department\_id, job\_id HAVING COUNT(\*) > 1 ORDER BY "How many" DESC;**

**A screenshot of a computer

Description automatically generated**

1. For each job id display the job id and total amount paid each month for this type of the job. Exclude job\_id *AD\_PRES* and *AD\_VP* and also include only jobs that require more than $15,000 in total.

Sort the output so that top paid jobs are shown first.

**SELECT SUM(salary), job\_id FROM employees WHERE job\_id <> 'AD\_PRES' AND job\_id <> 'AD\_VP' GROUP BY job\_id HAVING SUM(salary) > 15000 ORDER BY SUM(salary) DESC;**

A screenshot of a computer

Description automatically generated

1. For each manager number display how many persons he / she supervises. Exclude managers with numbers 100, 101 and 102 and also include only those managers that supervise more than 2 persons.

Sort the output so that manager numbers with the most supervised persons are shown first.

**SELECT manager\_id, COUNT(\*)**

**FROM employees**

**WHERE manager\_id NOT IN (100, 101, 102)**

**GROUP BY manager\_id HAVING COUNT(\*) > 2**

**ORDER BY COUNT(\*) DESC;**

A screenshot of a computer

Description automatically generated

1. For each department show the latest and earliest hire date, but exclude departments 10

and 20 and also exclude those departments where the last person was hired in this century. Sort the output so that most recent latest hire dates are shown first.

**SELECT department\_id, MIN(hire\_date), MAX(hire\_date)**

**FROM employees**

**WHERE department\_id NOT IN (10,20)**

**GROUP BY department\_id HAVING MAX(hire\_date) < '01-JAN-00'**

**ORDER BY MAX(hire\_date) DESC;**