# Lab 3 – (Sub-Select)

This week’s lab continues using the SELECT command in addition to now incorporating multiple tables in the FROM statement to gather information together.

## Getting Started

***Your submission will be a single text-based SQL file with appropriate header and commenting. Please ensure your file runs when the entire file is executed in SQL Developer.***

## Tasks

***It is important that Steps 1, 2, and 3 are completed first.***

1. Create an INSERT statement to do this. Add **yourself** as an employee with a NULL salary, 0.21 commission\_pct, in department 90, and Manager 100. You started TODAY.
2. Create an Update statement to: Change the salary of the employees with a last name of Matos and Whalen to be 2500.
3. Make sure you run a commit statement after the first 2 steps to make those changes permanent.

You **must use subqueries** for these questions (must minimize the number of tables being used in the main query, some of these can be solved using advanced join statements, but that is not the point of this lab)

1. Display the last names of all employees who are in the same department as the employee named Abel.
2. Display the last name of the lowest paid employee(s)
3. Display the city that the lowest paid employee(s) are located in. CAN BE DONE WITHOUT JOIN
4. Display the last name, department\_id, and salary of the lowest paid employee(s) in each department. Sort by Department\_ID. (*HINT:* careful with department 60) CAN BE DONE WITHOUT JOIN
5. Display the last name of the lowest paid employee(s) in each city REQUIRES A JOIN
6. Display last name and salary for all employees who earn less than the lowest salary in ANY department. Sort the output by top salaries first and then by last name.
7. Display last name, job title and salary for all employees whose salary matches any of the salaries from the IT Department. Do NOT use Join method. Sort the output by salary ascending first and then by last\_name