ReadMe file

**Project structure**

I’ve created a folder called “Tasks” which contains each single task as a standalone .sql file which can be run separately.

I have also created an “SQL\_Lab.sql” which contains all 5 tasks that can be run by clicking on the “Run sql file (Press twice)” button.

**Task 1:**

**Solution implementation**

For the first task I used the “SELECT” command to select the data I want from the database. In this case I would like to retrieve three data points which are

* ENAME – employee name
* DNAME – department name
* SAL – salary amount

So, I do “SELECT ENAME, DNAME, SAL”

Next, I used the “FROM” command to specify where is this data is coming from. In which table does this data exist in? In my case I said that this data exists in two places “EMP” employee table and “DEPT” department table.

So, I use “FROM EMP e, DEPT d”. In this case I use the “JOIN” command without actually typing the keyword “JOIN” which allows me to join both tables together.

Finally, I use the “WHERE” command to add the condition of my query.

So I do “WHERE SAL BETWEEN 1000 AND 2000 AND e.DEPTNO = d.DEPTNO;”

“SELECT ENAME, DNAME, SAL FROM EMP e, DEPT d WHERE SAL BETWEEN 1000 AND 2000 AND e.DEPTNO = d.DEPTNO;”

Putting everything together the sql query says the following

Please get me the column employee name, department name and salary from the employee table and department table. Please join those two tablets together and give me all employees in which their department number in the employee table match the department number in the department table which have a salary between 1000 and 2000.

**Alternative approach**

An alternative approach to doing this is not using the “BETWEEN” keyword but rather use the “>=” sign and “<=” sign like so

“SELECT ENAME, DNAME, SAL FROM EMP e, DEPT d WHERE SAL >= 1000 AND SAL <= 2000 AND e.DEPTNO = d.DEPTNO;”

**Task 2:**

**Solution implementation**

For the second tasks I have used two commands to retrieve how many people receive a salary vs a commission in department 30.

“SELECT COUNT(\*) FROM EMP WHERE DEPTNO = 30 AND SAL >= 0;”

This query says please get me all the records in employee table where the department number is 30 and the salary is larger than zero. This query will return everyone in department 30 that get a salary.

The reason I did bigger than or equal to zero is because I consider the value zero as receiving a salary even though the amount is zero. If the value was null on the other hand, then that means this employee does not receive any salary.

“SELECT COUNT(\*) FROM EMP WHERE DEPTNO = 30 AND COMM >= 0;”

This query is the same as above except now we are getting employee that receive a commission rather than a salary.

**Alternative approach**

I am not sure if there is an alternative way to do this.

**Task 3:**

**Solution implementation**

For the third task I used the below query to retrieve all employee names and salaries that work in Dallas.

“SELECT ENAME, SAL FROM EMP e, DEPT d WHERE e.DEPTNO = d.DEPTNO AND d.LOC = 'Dallas';”

The above query says please select employee name and salary. Please join the employee and department table into one and get me every employee with a employee department number that matches the department number and employees with a location equal to Dallas.

**Alternative approach**

I am not sure if there is an alternative way to do this.

**Task 4:**

**Solution implementation**

For the fourth task I have to list all employees that do not belong to any department. The way I accomplish that is by the below query

“SELECT \* FROM DEPT WHERE DEPTNO NOT IN (select DEPTNO FROM EMP);”

The above query says please select every column in the department where the department number is not in the employee table.

**Alternative approach**

You could join both the department table and employee table using the join keyword

**Task 5:**

**Solution implementation**

The fifth and final task is to list the average salary of every department. The way I accomplish that is by the below query.

“SELECT DEPTNO, AVG(SAL) FROM EMP GROUP BY DEPTNO;”

The above query says please select all the department numbers and get the average salary from the employee table by using the AVG function and group everything by the department number.

**Alternative approach**

You could join both the department table and employee table using the join keyword