

HW 3 (SQL)

DSCI 551, Fall 2022

100 points

Due: 10/21, Friday, 11:59pm

Install employee database as described in:

<https://dev.mysql.com/doc/employee/en/employees-installation.html>

You can either download the database zip file from https://github.com/datacharmer/test_db or use the zip file we downloaded and include in this homework folder. The zip file is: test_db-master.zip.

To install the database on EC2:

1. Download the test_db-master.zip file
2. sftp it to your EC2 instance.
3. On your EC2, execute:
 unzip test_db-master.zip
 cd test_db-master
 mysql -u root -p < employees.sql
4. Now log in to your mysql database. You should see a new database called employee now.

In this homework, ignore the views `current_dept_emp` and `dept_emp_latest_date` in the database.

A. [50 points] Write an SQL query for each of the following questions on the above employees database. Note your query should run without error in MySQL.

1. Find employee records for employees whose first name contains 'mary' and the second to the last letter in the last name is 'o'. Note like operator is NOT case sensitive.

format of output: +-----+
 | emp_no |
 +-----+

2. Find first name and last name of employees who ever made at least \$150000. Show the same employee (with the same name) only once.

format of output: +-----+-----+
 | first_name | last_name |
 +-----+-----+

3. Find out for each department, how many employees have worked for the department with a start date of '1988-10-20'. Show the department number and the number of employees in the descending order of the number of employees.(Do not care about the employee still works in the department or not)

format of output: +-----+-----+
 | dept_no | employeeNum |
 +-----+-----+

4. Find the department number (dept_no) of departments which have or had at least three different managers.

format of output: +-----+

dept_no

- Find first name, last name, and title of employees where the title contains 'engineer', e.g., 'senior engineer' or 'engineer' for the position starting from '2000-3-23'. (Do not care about the employee still works in the department or not)

format of output:

first_name	last_name	title

For Question B and C, you will use the beers-tables database in the lectures.

B. [30 points] Write an SQL query for each of the following questions on the beers-tables database. Note your query should run without error in MySQL.

- Find manufacturers who made at least three different beers. Your column names should look EXACTLY like:

Manufacturer
- Find drinkers who never frequent bars. Your column names should look EXACTLY like:

Drinker
- Find out for each bar, how many beers are sold at the bar with a price of at least 2 dollars. Your column names should look EXACTLY like:

Bar	Total
- Find the bars who sell the most expensive beers. Your column names should look EXACTLY like:

Bar
- Find drinkers who like Bud but do not like Summerbrew. Your column names should look EXACTLY like:

Drinker

C. [20 points]

- Create a view called Beers2Bars that lists the manufacturer, beer, bar, and price for each beer sold at the bar.

2. Write a query that uses only the above view to find the average price of beers for each manufacturer. Your column names should look EXACTLY like:

```
+-----+-----+
| Manufacturer | Average |
+-----+-----+
```

**** Submission requirements ****

PLEASE READ CAREFULLY BEFORE SUBMISSION.

1. For each question, submit a .sql file with SQL query (or create view statement) and the result of the query. You can use any text editor to save the file.

Note that you should design a general-purpose query to answer each question. For example, if the data in the beers-tables database are slightly changed, your command must still be able to get the correct results.

2. File format must be: *.sql

e.g. A1.sql, B1.sql, C1.sql, ... (12 files in total)

DO NOT submit any other files that are not required.

DO NOT submit a zip file.

3. Make sure you follow the correct “multi-line comment” format of in sql file with /* */.

DO NOT write anything except SQL commands or statements outside of comments. For output results, you can paste the content inside /* */, e.g.:

```
select * from Beers;
```

```
/* Output
```

```
+-----+-----+
| name   | manf   |
+-----+-----+
| Bud    | Anheuser-Busch |
| Bud Lite | Anheuser-Busch |
| Budweiser | Heineken      |
| Michelob | Anheuser-Busch |
| Summerbrew | Pete's       |
+-----+-----+
5 rows in set (0.00 sec)
*/
```

4. You may submit multiple times, but only the last version will be saved. All previous submissions will be overwritten automatically. So make sure to submit all of your files.