

CSE 4/560 Project 2: Company DB - SQL Query

Due 23:59 04/19/2020 EST

April 15, 2020

This is an individual project for writing SQL queries. There is 14 problems with 25 points in total. **Please note that academic integrity is strictly implemented and any violation will lead to a F grade in this course.**

1 Project Setup

1.1 MySQL

This project ONLY use MySQL (version 8.0.13) as the canonical database. To download MySQL community server, please go to <https://downloads.mysql.com/archives/community/>.

1.2 Database: Employees

Follow the steps below to install the project database

1. Download the GitHub Repository: https://github.com/datacharmer/test_db
2. Launch command line console, change the working directory to your downloaded repository
3. Type following command:
`mysql < employees.sql`
or
`mysql -u YOUR_MY_SQL_USER_NAME -p < employees.sql`
This will initialize your database.
4. To verify installation, run following commands:
`mysql -t < test_employees_md5.sql`
or
`mysql -u YOUR_MY_SQL_USER_NAME -p < test_employees_md5.sql`

2 Problem Statements

For each problem, write a SQL query to find the information described in the statements. Only following MySQL functions are allowed during constructing SQL Queries:

- AVG
- COUNT
- DATEDIFF
- MIN
- YEAR

Answer Format: Each problem can only have ONE SQL query. The query can be arbitrary complex, such as nested query etc. Write the query in a file with letter q followed by the problem number and .sql extension. e.g., the answer query for problem 1 is written in q1.sql as file name.

2.1 Problem 1, 1 point

Find all employees' employee number, birth date, gender. Sort the result by employee number. The result of query is similar to following table:

emp_no	birth_date	gender
10001	1953-09-02	M
10002	1964-06-02	F
10003	1959-12-03	M
...		

2.2 Problem 2, 1 point

Find all female employees and sort the result by employee number. The result of query is similar to following table:

emp_no	birth_date	first_name	last_name	gender	hire_date
10002	1964-06-02	Bezalel	Simmel	F	1985-11-21
10006	1953-04-20	Anneke	Preusig	F	1989-06-02
...					

2.3 Problem 3, 1 point

Find all employees' last name with their salaries in different periods. Sort the result by last name then salary. The result of query is similar to following table:

last_name	salary	from_date	to_date
Aamodt	39537	1991-05-28	1992-05-27
Aamodt	39548	1986-12-31	1987-12-31
...			
Acton	39202	1994-10-10	1995-10-10
Acton	39581	1993-10-10	1994-10-10
...			

2.4 Problem 4, 1 point

Find all employees' current department and the start date with their employee number and sort the result by employee number. The result of query is similar to following table:

emp_no	dept_name	from_date
10001	Development	1986-06-26
10002	Sale	1996-08-03
10003	Production	1995-12-03
...		

2.5 Problem 5, 1 point

List the number of employees in each department. Sort the result by department name. The result of query is similar to following table:

dept_name	noe
Customer Service	23580
Development	85707
...	

2.6 Problem 6, 2 points

List pairs of employee (e_1, e_2) which satisfies ALL following conditions:

1. Both e_1 and e_2 's current department number is d001.
2. The year of birthdate for e_1 and e_2 is 1955.
3. The e_1 's employee number is less than e_2 .

Sort the result by e_1 then e_2 . The result of query is similar to following table:

e1	e2
10239	10367
10239	11251
...	
10367	11251
10367	11554
...	

2.7 Problem 7, 2 points

For each department, list out the manager who stayed the longest time in the department. The list needs to exclude the current manager. Sort the result by employ number. The result of query is similar to following table:

emp_no	dept_name
110022	Marketing
110085	Finance
...	

2.8 Problem 8, 2 points

Find out departments which has changed its manager more than once then list out the name of the departments and the number of changes. Sort the result by department name. The result of query is similar to following table:

dept_name	cnt
Customer Service	3
Development	1
...	

2.9 Problem 9, 2 points

For each employee, find out how many times the title has been changed without changing of the salary. e.g. An employee promoted from Engineer to Sr. Engineer with salaries remains 10k. Sort the result by employ number. The result of query is similar to following table:

emp_no	cnt
10004	1
10005	1
10007	1
10009	2
...	

2.10 Problem 10, 2 points

Find out those pairs of employees (e_H, e_L) which satisfy ALL following conditions:

1. Both e_H and e_L born in 1965
2. e_H 's current salary is higher than e_L 's current salary
3. e_H 's hiring date is greater than e_L , which means e_H is a newer employee than e_L .

Sort the result by employee number of e_H then employee number of e_L . Result is shown as table below:

h_empno	h_salary	h_date	l_empno	l_salary	l_date
10095	80955	1986-07-15	17206	55078	1986-02-25
10095	80955	1986-07-15	18617	66957	1986-06-28
...					

- h_empno : e_H 's employee number
- h_salary : e_H 's current salary
- h_date : e_H 's hire date
- l_empno : e_L 's employee number
- l_salary : e_L 's current salary
- l_date : e_L 's hire date

2.11 Problem 11, 2 points

Find the employee with highest current salary in each department. Note that MAX function is not allowed. Sort the result by department name. Result is shown as table below:

dept_name	emp_no	salary
Customer Service	18006	144866
Development	13386	144434
...		

2.12 Problem 12, 2 points

Calculate the percentage of number of employees' current salary is above the department current average. Sort the result by department name. The result is shown as following:

dept_name	above_avg_pect
Customer Service	47.1455
Development	51.9825
...	

As the figure shows, there are 51.9825 % employees in Development department has their current salary above the average of current salary in Development department.

2.13 Problem 13, 3 points

Assuming a title is a node and a promotion is an edge between nodes. e.g. And promotion from Engineer to Senior Engineer means there is a path from Node 'Engineer' to Node 'Senior Engineer'. Find out pairs of node of source and destination (src, dst) which there is no such path in the database. Sort the result by src then dst. The result is shown as following:

src	dst
Assistant Engineer	Assistant Engineer
Engineer	Assistant Engineer
...	

The result table shows that there is no path from Assistant Engineer to Assistant Engineer and neither Engineer to Assistant Engineer. That means there is no one have been from Engineer and be promoted/demoted to Assistant Engineer (no matter how many times of promotion/demotion) in the database.

2.14 Problem 14, 3 points

Continued from problem 13, assumeing we treat the years from beginning of a title until promotion as the distance between nodes. e.g. An employee started as an Assistant Engineer from 1950-01-01 to 1955-12-31 then be promoted to Engineer on 1955-12-31. Then there is an edge between node "Assistant Engineer" to "Engineer" with distance 6.

Calculate the average distance of all possible pair of titles and ordered by source node. To simplify the problem, there is no need to consider months and date when calculating the distance. Only year is required for calculating the distance. Besides, we can assume the distances of any given pair is less than 100.

Sort the result by src then dst. The expected result is shown as follow:

src	dst	years
Assistant Engineer	Engineer	7.7926
Assistant Engineer	Manager	20.5266
...		
Engineer	Manager	12.7340
...		

As the table shows, the average distance between node "Assistant Engineer" and node "Engineer" is 7.7926. We add it with the distance between "Engineer" to "Manager", which is 12.7340, to find out the distance between "Assistant Engineer" to "Manager" is 20.5266.

3 Offline Grader

Before downloading and using the offline grader, please pay attention to following points:

1. The grader strictly compares the EXACTLY same result and order mentioned in each problem statement.
2. The grader checks DB state on start, make sure the DB state is same as the state which is immediately after importing the employees database.

3. The grader takes the query run time into account, you might get partial or no point if the query is running too slow.
4. The score is unofficial, we will run the grader with your submission after project due date as the official score.

The grader only supports Windows and Mac operating system. After downloading the zip file, follow the instructions according to the platform.

3.1 Windows

1. Make sure mysql server is running on localhost.
2. Decompress the zip file, the result is a directory named *proj2-grader-win*
3. Edit the *proj2.cfg*, set the user and password for the mysql server connection.
4. Launch a console such as cmd or powershell, change the working directory to *proj2-grader-win*
5. Execute *proj2_test.exe* from console, the result should be a pass on initial state verification and failed on all questions.
6. Write your answer in the files in *quiz* directory, each question has one file. e.g., writing the answer for problem 1 in *q1.sql*
7. Run *proj2_test.exe* again, grader will show the scores.

3.2 Mac OS X

1. Make sure Python 3 is installed at `/usr/local/bin/python3`
2. Make sure mysql server is running on localhost.
3. Decompress the zip file, the result is a directory named *proj2_test.app*
4. Launch a console, change the working directory to *proj2_test.app/Contents/Resources*.
5. Edit the *proj2.cfg*, set the user and password for the mysql server connection.
6. Change the working directory to *proj2_test.app/Contents/MacOS*
7. Execute *proj2_test* from console, the result should be a pass on initial state verification and failed on all questions.
8. Write your answer in the files in *proj2_test.app/Contents/Resources/quiz* directory, each question has one file. e.g., writing the answer for problem 1 in *q1.sql*
9. Run *proj2_test* again, grader will show the scores.

4 Submission

Failure to comply with the submission specifications will incur penalties for EACH violation.

- What to submit: A zip file has to be submitted through the ‘submit_cse460’ (if you are CSE460 student) or ‘submit_cse560’ (if you are CSE560 student) submit script by 04/19/2020 11:59PM EST. Only zip extension will be accepted, please **don’t** use any other compression methods such as tar or 7zip. You can submit multiple times, note that **only** the last submission will be kept on the server.
- Zip file naming: Use *ubit_proj2* (**NO SPACE!**) for the filename, for example: *jsmith_proj2.zip*, where *jsmith* is the ubit of submitter. The project is an **INDIVIDUAL** project, so everyone needs to submit **ONE** zip file.
- Sub-structure of zip file: On unzipping the zip file, there should be a folder named with your ubit *ubit_proj2*, under the folder *ubit_proj2*, there should be 14 SQL files, starting from *q1.sql*, *q2.sql* ... , *q14.sql* which correspond to SQL query for each problem.