CSC3170 Introduction to Database Systems (Fall 2022)
Assignment 3

Please answer all the questions below and submit your answer to blackboard ***on or before 27th November 2022 23:59***

(Note: late submission will lead to deduction of marks or even zero points.)

1. Background

A software company sells and develops software applications. The company has many offices and each office has a unique ID. Each engineer is assigned to one of the offices. An engineer has his own assignments, such as conducting research on a project, or maintaining a software application (it is possible for him to have multiple assignments). Each project belongs to a category and has its own budget and expenditure. Each software applications must be associated with one project.

2. Schema

OFFICE (OFFICEID, ONAME, OADDRESS, OPHONENUM)
ENGINEER (ENGINEERID, ENAME, ESALARY, OFFICEID)
PROJECT (PROJECTID, OFFICEID, PNAME, PCATEGORY, PBUDGET, PEXPENDITURE)
APPLICATION (APPID, ANAME, APRICE, ADATE, PROJECTID)
MAINTAIN (APPID, ENGINEERID)
RESEARCH (PROJECTID, ENGINEEERID)

3. Description

OFFICE

Field Name	Format	Description
OFFICEID	Integer	The ID of the office. It is the primary key.
ONAME	255 Characters	The name of the office.
OADDRESS	255 Characters	The address of the office.
OPHONENUM	100 Characters	The phone number of the office.

ENGINEER

Field Name	Format	Description
ENGINEERID	Integer	The ID of the engineer. It is the primary key.
ENAME	255 Characters	The name of the engineer.
ESALARY	Integer	The salary of the engineer.
OFFICEID	Integer	The ID of the office that the engineer belongs to.

PROJECT

Field Name	Format	Description
PROJECTID	Integer	The ID of the project. It is the primary key.
OFFICEID	Integer	The ID of the office that holds the project.
PNAME	255 Characters	The name of the project.
PCATEGORY	255 Characters	The category of the project.
PBUDGET	Integer	The budget of the project.
PEXPENDITURE	Integer	The expenditure of the project

APPLICATION

Field Name	Format	Description
APPID	Integer	The ID of the application. It is the primary key.
ANAME	255 Characters	The name of the application.
APRICE	Integer	The price of the application.
ADATE	Integer	The release date of the application. Stored as a 8-digit integer of the form yyyymmdd.
PROJECTID	Integer	The ID of the project that this application is based on.

MAINTAIN

Field Name	Format	Description
APPID	Integer	The ID of the application. It is a foreign key referred to APPID in APPLICATION
ENGINEEERID	Integer	The ID of the engineer. It is a foreign key referred to ENGINEEERID in ENGINEEER

RESEARCH

Field Name	Format	Description
PROJECTID	Integer	The ID of the project. It is a foreign key referred to PROJECTID in PROJECT
ENGINEEERID	Integer	The ID of the engineer. It is a foreign key referred to ENGINEEERID in ENGINEEER

4. Queries

Based on the above information, write the following queries in SQL statements on the provided MySQL database account:

1. Find the number of the engineers who work at the office named “Database Group” and whose salary is smaller than 40000.

2. Find the **ENGINEERID** and **ENAME** of the engineer(s) who are currently working on every project under category “Web”. The result should be sorted by the **ENGINEERID** of the engineer in ascending order. The ordering of the columns is shown below:

ENGINEERID	ENAME

3. Find the **ENAME** of the engineer(s) who have the highest salary at the office named “Network Coding Research Center”. The result should be sorted by **ENAME** in descending order.

4. Find the **ANAME** and **APRICE** of the cheapest 4 applications maintained by engineer named ‘Jane Xiu’. The result should be sorted by **APRICE** in descending order. The ordering of the columns is shown below: (Assumption: The price of each application maintained by ‘Jane Xiu’ is unique.)

ANAME	APRICE

5. Find the **APPID** and **ANAME** of application(s) who are maintained by more than 2 engineers. The result should be sorted by **APPID** in ascending order. The ordering of the columns is shown below:

APPID	ANAME

6. Find **OFFICEID** and **ONAME** of the office(s) of which the total budget of all the projects is larger than 30000. The result should be sorted by **OFFICEID** in ascending order. The ordering of the columns is shown below:

OFFICEID	ONAME

7. The **HEAD COUNT** of a project is the number of engineers working on it (through the research relation only). Now, for every project category, calculate the average **HEAD COUNT** of the projects for that category. The result should be sorted by the **PCATEGORY** name. The ordering of the columns is shown below: (Note: You may assume that every project must have at least one engineer working on it and you may need to use **AVG()** in your query. Please refer to https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_avg)

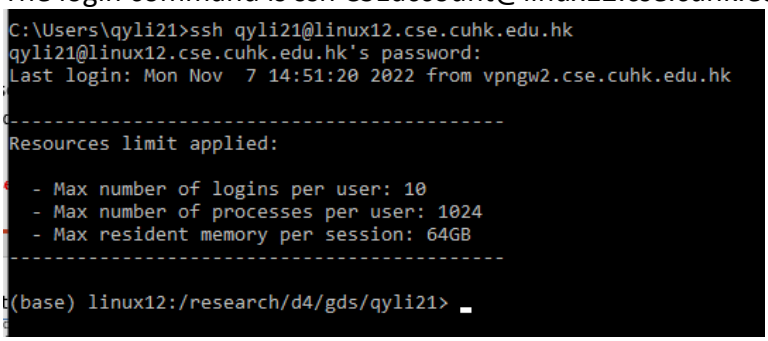
PCATEGORY	HEAD_COUNT

8. Find the **ENGINEERID** and **ENAME** of the project engineer who is working on the project(s) with the highest budget surplus (i.e. PBUDGET - PEXPENDITURE). The result should be sorted by **ENGINEERID** in ascending order. The ordering of the columns is shown below: (Note: You may need to perform arithmetic operation in your query. Please refer to https://dev.mysql.com/doc/refman/5.5/en/arithmetic-functions.html)

ENGINEERID	ENAME

5. Connection to Oracle DB

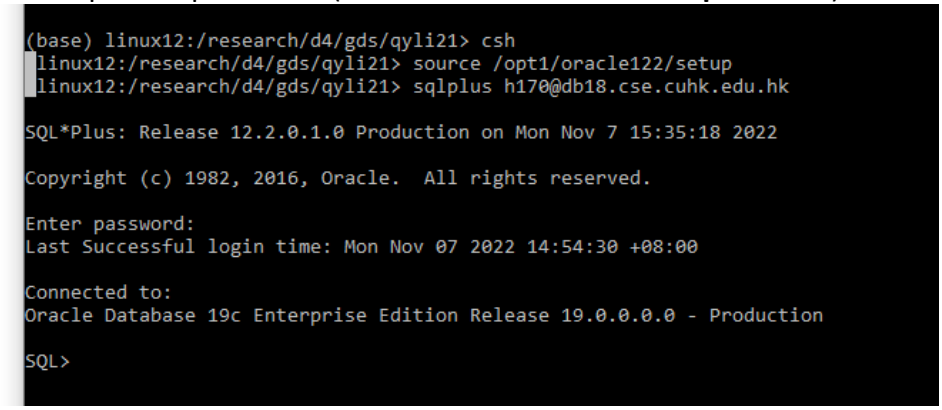
You should first connect to the linux host with your CSE account, e.g. linux 12 (Please choose linux 10-15. If you don't have a CSE account, please contact qyli21@cse.cuhk.edu.hk).
The login command is ssh CSEaccount@linux12.cse.cuhk.edu.hk



Before you connect to the database, please first transfer the data folder to the server and enter the data fold. After that, type

- csh
- source /opt1/oracle122/setup
- sqlplus oracle_account@db18.cse.cuhk.edu.hk

Then input the password. (Please note **the case of the password**)



The whole assignment 3 is based on terminal. Even if you do this assignment based on the SQL workbench, please submit your final answer based on the terminal, otherwise, the Spool command will not work.
If you have successfully connected to the sql server through the terminal, you can then create the tables and add data entries using the files in the data fold. First, type '@create_table.sql' in the terminal and then type '@add.sql' in the terminal. After that, the database is created successfully and you can complete your assignment now.

6. Submission Procedure

You should **STRICTLY** follow this procedure to submit all your SQL queries or you may receive mark deduction. Assume your name is "Chan Tai Man" and your student ID is 1101234567. The submission procedures are shown as follows:

1. Write your queries to single file called **<your_student_ID>.sql** (e.g. 1101234567.sql) for all of the above queries and save the query results to the files result1.lst, result2.lst, …, result8.lst for queries 1, 2, …, and 8 respectively using the Spool command in MySQL (see the example shown below).
You should use comment lines to include your name and student ID at the header of 1101234567.sql. You should also use the Oracle command tee for each of the queries. Do NOT add any comment lines inside your SQL statements. There is always at least one space between your comment body and /* (or */). Your 1101234567.sql should be in the following format:

```
/*
    Student ID: 1101234567
    Name: Chan Tai Man */
/* Query 1 */
Spool result1.lst
Select … from … ;
Spool off
/* Query 2 */
Spool result2.lst
Select … from … ;
Spool off
……
If you need to create views, DO NOT write the create and drop sql
statements inside
the body of Spool. The format should be:
/* Query 8 */
Create OR Replace view temp AS …
Spool result8.lst
Select … from … ;
Spool off
Drop view temp;
```

Please use a Unix text editor instead of a Windows editor, or you should ensure that your submitted file should not contain any special characters (e.g. ^M), which are resulted from transferring your files from Windows to Unix, by using a Unix command dos2unix on linux machine.

You should test your final .sql file (e.g. 1101234567.sql) **before submission** by typing the command "@<your_student_ID>.sql;" (e.g. @1101234567.sql;) in your Oracle account. This should generate the result files result1.lst, result2.lst, …, result8.lst in your current directory in Unix. You have to ensure that the content of each result file is correct in order to get score for the query.

2. Submit your .sql file to the submission box on the blackboard.