

Database Systems, CSCI-GA.2433-011, New York University, Fall 2020

Instructor: Professor X. Sean Wang ([x.sean.wang@nyu.edu](mailto:x.sean.wang@nyu.edu))

## Homework Assignment #2

**Assignment Due time: Nov 4, 2020 @ 23:00 (CST)**

There are **10 queries to write**. You need to submit **1 file** to the Classes site. **Read the whole assignment carefully.**

**What to SUBMIT: one MySQL script file. Naming for the file:** Assuming that your Net ID is abc123 and you are submitting your solution to Homework due 2034-02-15, your file should be named 20340215abc123.sql (of course you need to specify the correct date and the correct Net ID). **Repeated submission:** repeated submission on Classes site is allowed as long as it's before due time (and while the submission site remains open), and the **last submission** will be looked at for grading.

**Team work with another student is allowed for this assignment.** If the work is done collaboratively by two students, one of the students should submit the script file as above but with the name and netID of both students in the files (use # at the beginning of a line to add comments to script file), and the other student submit simply a text file, indicating the team work by listing the name and netID of both students. Note in this case, both students will receive the same score.

See <https://cims.nyu.edu/webapps/content/systems/userservices/databases> for instructions on how to set up MySQL database on CIMS. I recommend using "self-administered MySql database"

(<https://cims.nyu.edu/webapps/content/systems/userservices/databases-selfmanaged>)

For this assignment, you are to create a database (using MySQL) with the following tables with the data as follows.

Employee(Name, Salary, Manager, Department)

Smith, 31000, Jones, Switch  
Patten, 28000, NULL, Software  
Hughes, 33000, Jones, Switch  
Jones, 32000, Patten, Switch  
Warren, 40000, Patten, Software

Key is Name.

Course(Student, Subj, Prof, Grade)

Smith, Algs, Hackett, 85  
Patten, Algs, Hackett, 80  
Patten, Comp, Roe, 70  
Jones, Comp, Roe, 75  
Jones, Dbase, Black, 80  
Warren, Dbase, Black, 75  
Warren, Comp, Roe, 65

Key is Student, Subj.

Write all your SQL queries in one script (text) file and submit the script file (see above). The script file (text format) simply lists all the SQL queries you have as answers for the problems below (separated by semicolon, certainly). Aside from reading your answers, we will run the script file on several similar but different databases (modified from the example data above. Remember: a query does not just work on one instance of the database!). Hence your script file should be runnable from MySQL directly, for example, from mysql prompt as follows:

```
mysql> source scriptfile.sql
```

Each query is worth 10 points.

1. Find names of all employees who work in the software department. On above example data, should be Patten and Warren.
2. Find names of all employees who earn at least 5000 more than their managers. On above example data, should be Warren.
3. Find names of all employees who received higher grades than their managers in the same course. On above example data, it should be Jones.
4. Find name of all employees who do not take any class. On above example data, should be Hughes.
5. Find departments in which all of their employees take (one or more) courses. On above example data, it should be Software. *Hint: it's fine to use temporary tables.*
6. Find departments in which all of their employees take two or more courses. On above example data, it should be Software. *Hint: it's fine to use temporary tables.*
7. Find the average salary earned. On example data should be 32,800.
8. Find the lowest salary earned by people taking Roe's course. On example data, it should be 28,000.
9. For those departments whose employees (collectively) take more than three courses, find the average salary by department. On example data, it should be 34,000. *Hint: It's fine to use two queries and a CREATE TEMPORARY TABLE command that finds relevant departments first.*
10. For each department, find the percentage of the employees who do not take any course. On example data, it should be two tuples "(Software, 0), (Switch, 33.33)". Note that if all the employees of a department D take courses, then the department should get a result as "(D, 0)". For example, if your answer on example data is simply (Switch, 33.33) (without (Software, 0)), then your query is wrong. *Hint: you may use case function like "Case when C is null then 0 else C/D end" in the Select Clause.*

*General hint: The operation MINUS is not supported by SQL, which can be replaced via "left join". For example, the following MINUS operation:*

```
Select A From R1
MINUS
Select A From R2;
```

*may be replaced by:*

```
Select A
From R1 left join R2 on R1.A=R2.A
Where R2.A is NULL;
```