NYC Data Science Bootcamp

SQL

***** Save all your queries to *yourname*.sql and push it to the homework Github repository.

Note: you may need to pull from origin before you push it to Github.

Question #1:

For this question, use the sqlexercise database. There is a table called SLEEP.

1. Write a query to return columns id and extra from table SLEEP.
2. Rewrite the previous query so that extra will appear as the first column in your query result.
3. Write a query to return all the category values, without repetitions.
4. Write a query to return every id whose extra > 0.
5. Write a query to return the total of extra in each category (call it extraSum) and the number of records in each category (call it categoryNum).
6. Write a query to return the average extra of each category (call it mean_extra).

Question #2:

For the following questions, use your own database(*username_db*). There are two tables "Department" and "Employee". Answer the following questions using these two tables.

1. Select the first two rows in table Department.
2. Write a query to return the employeename, hiredate, basewage from table Employee.
3. Write a query to return the total wage of employees.

   total wage = basewage $*$ baselevel

4. Write a query to return names of employees whose basewage ranges from 2000 to 3000, sort the result by basewage in descending order. *(Look up online how to use ORDER BY to sort descending.)*

5. Write a query to return the employeename, hiredate, basewage whose name ends with 8 and who was hired after June 10, 2010. (Hint: read pattern matching in Mysql)

6. Write a query to return the employeename and corresponding departmentid whose total wage is larger than 7000.

7. Write a query to return the departmentid of departments that have at least 2 employees with basewage >= 3000.

8. Write a query to return the average total wage in each department. Sort the results by average wage in ascending order.

9. Write a query to return the average total wage of males and females in each department. Sort the results by DepartmentID in descending order.

10. Write a query to return the name of each employee, along with his/her deparmentname and the principal in the department. (*Hint: use JOIN.*)

Question #3:

1. Download this data set; the information about attributes is here (item 7). Upload it to the server using scp. Create a table named adult which has the same structure. Two ways to get the data on server:

    a. Download the data directly on our server using curl or wget;

    b. Download on your local machine and copy it to server using scp.

2. Load the data set adult.data into table adult (***Note**: In the original data, separators are ", "[a comma followed with a space], remember to set the value of* FIELDS TERMINATED *to ", " instead of ","*).

3. Are there any missing values in the table? How many rows have missing values?

    a. For numerical fields, use the '*is null*' condition.

    b. For string fields, missing values are represented as "?".

4. Remove the rows having missing values.

5. What's the ratio of *number of '<=50K' / number of '>50K'* in column class. (*Hint*:

Create two “temporary” tables.)

6. Compute the average age in each class.
7. How many rows in class ‘>50K' where the age is less than 36.78?
8. What's the average hours-per-week in each class?
9. What's the ratio of *number of '<=50K' / number of '>50K'* in Female and Male (column sex)? (*Hint*: Create two “temporary” tables.)