Database Systems, CSCI 4380-01
Homework # 6
Due Tuesday March 20, 2018 at 11 PM

Homework Statement.

This homework is worth 3% of your total grade. If you choose to skip it, Midterm #2 will be worth 3% more.

This homework is on writing procedural SQL and insert/update/delete statements. It requires you to take a complex task, break it into computationally inexpensive steps and put all of it into a single procedure. You are free to use any SQL construct, any number of steps and statements. There is no penalty for long or short solutions, just get the job done.

Write a script that creates a single procedure called `hw6()` that when called:

```
select hw6();
```

accomplishes the following data cleaning tasks:

- Adds three new columns `sport,discipline,stype` to the `events` table and populates these fields with the corresponding information from the sports table (use update statement).

  While this process is a total opposite of normalization (actually called denormalization), sometimes it is desirable to avoid joins. Since the sports relation rarely changes, this is not a big problem.

  To add a column to an existing table, you can use the following command (a seperate command for each column):

  ```
  alter table events add sport varchar(255);
  ```

- Add a column `category` to the `events` table and copy all event categories (`winter_eventcategories.etype)` (use update statement). Remember that categories are for a specific event id, but they hold for any event with the same `name`, `discipline` and `etype`.

  While `sports` and `winter_eventcategories` tables are no longer needed, you do not need to drop them.

- There are multiple events with the same `name`, `discipline` and `etype`, but different id values. Merge all these events so that the `events` table contains only one tuple for each event `name`, `discipline` and `etype` (delete tuples corresponding to duplicates events).

  However, when doing this you must make sure that the medal information for summer and winter medals is not lost. Furthermore, the categories from the `winter_eventcategories` table are transferred correctly. In short, queries before and after this step should return the same information except for the ids.

If you think it is helpful, you can create secondary tables and use them within the function. However, you must drop all these tables before your function ends.

Data for this homework

The data for this homework is a down sampled version of the homework #5 data. Basically, I deleted almost 95% of the data to make it easy to test your code. You must write efficient code. I will test your code with different and larger data. Unfortunately, I cannot provide you with more data because the class server does not work well with large data scripts. If you have your own server, you can test it with the Hw#5 data.

I will provide some test queries for testing the effects of your function later in the week.

Using class server for this homework

If you are using the class server at `http://rpidbclass.info` for this homework, you must connect to your own database, named `db_yourusername`. Here you can create tables and functions, modify the data. If you find that you need to restart, simply recreate the database using the data scripts provided to you. Simply cut and paste the contents of the whole script into the SQL Command window (first table creation and then the data insertion).

Do not create any functions or tables in shared databases.

Use only the SQL Command interface for testing and data creation. Other methods seem to freeze the database.

Submission Instructions.

Submit a single script that create a single function called `hw6()` with no inputs. Do not create any other functions or tables. This will make testing very difficult. Your script should create the function, but not execute it.

Your function should accomplish all the data cleaning described and not leave any additional tables when completed. It is not important whether it returns something or not. I am providing you with a skeleton function that you can use and add to.

Submit your script in a PDF on gradescope. You can use the template I posted for Hw#4.

Also, you must submit the Ascii text content of your script to `lms.rpi.edu`. Unfortunately your function has to have `$$` as delimiters, so we cannot use the same method as homeworks #4 and #5. So, we will try instead `lms.rpi.edu` for this purpose.

Sorry for all the changes in the submission method. Hopefully this will be an easy to use system.