CS 4400X: Introduction to Database Systems Spring 2022

Assignment 2

Instructor: Xu Chu *Due: Mar 9 at 11:59pm EST*

Submission instructions: Submit your submission.py file via Canvas.
TA: Peng Li is the responsible TA for A2. All questions regarding A2 (grading, clarification) shall be addressed by visiting TA office hours or contacting Peng at pengli@gatech.edu.

1 SQL (100 Points)

In assignment 1, you have gotten familiar with BigQuery WebUI. In addition to WebUI, BigQuery also provides many client libraries, which enables us to run queries with our favorite programming languages. Let's try BigQuery Client Python Libraries. Read and understand **BigQuery-Python Tutorial** (availabel on canvas). Follow the instructions in the tutorial to load dataset *cs4400_ncaa_basketball* to your own project. The relational schema of this dataset is shown below.

- player(id, name, birthplace_city)
- team(id, name, school, venue_id, color)
- game(id, season)
- venue(id, venue_city, venue_capacity)
- team_game(game_id, h_id, h_points, a_id, a_points)
- player_game(game_id, player_id, team_id, points)

cs4400_ncaa_basketball dataset is a simplified version of public NCAA Basketball dataset. The *team_game* relationship stores the information of two teams (home team and away team) for each game. *h_id* and *a_id* indicate the id of home team and away team respectively. *h_points* and *a_points* indicate points that home team and away team scored respectively in this game. The *player_game* relationship stores the information of players for each game. *team_id* is the id of the team that the player played for in the game. A player might transfer to another team and played for different teams in different games. *points* indicates the points that the player scored in the game. *points* can be null, which indicates the player enrolled but not played in the game. All other relations should be self-explanatory.

What to submit: Write SQL for each of the following queries about *cs4400_ncaa_basketball* dataset. Copy and paste all the queries you wrote into **submission.py** file (available under A2 folder). I would recommend you to run queries first using WebUI, where you can check your SQL syntax and have a preview of the results. Then copy the SQL into corresponding place in **submission.py** and run queries using python as described in the tutorial to double check the results.

(1) List players who enrolled in games in both season 2013 and season 2017. The output format should be (id, name) and ordered by name in alphabetical order.

(2) List players who enrolled in at least 40 games in a season. The output format should be (id, name) and ordered by name in alphabetical order.

(3) List players who scored more than 10 points in every game he played and who has played at least 10 games. The output format should be (id, name) and ordered by name in alphabetical order. Hint: A player played a game if his score is non-null.

(4) List players who have played games in the same city where they were born. The output format should be (id, name) and ordered by name in alphabetical order. Hint: A game is played in the venue of the home team and a player played a game if his score is non-null.

(5) We define partners as a pair of players who played (i.e. two players have non-null points) as teammates in a game and partner_points as the sum of points that two player scored in the game. List partners who played in at least 30 games together in season 2017 and the average partner_points they scored in those games are greater than 40. The output format should be (player1_id, player2_id, average_partner_points), where player1_id $<$ player2_id. Sort your output by average_partner_points from highest to lowest and round the average_partner_points to 2 decimal places.

(6) What teams have the maximum possible red intensity in their color? The output format should be (team_name, school, color) and ordered alphabetically by the team name. Hint: Hexadecimal colors codes are a way of representing color on a computer. Hex color codes are of form #AABBCC, where AA, BB, and CC are hexadecimal numbers (00, 01, ... , FE, FF) indicating the intensity of red, green, and blue in the color, respectively. Be careful with the case of the colors in the dataset – some use lower case characters and some use upper case characters.

(7) How many home games Georgia Tech's basketball team won and lost in season 2017? The output format should be (num_win, num_lose).

(8) What is the top 10 biggest margin of victory? Output the winning team name, losing team name, winning team points, losing team points, and the win margin of the games. In case of tie in the margin, sort the result by winning team points. The format should be (win_team_name, lose_team_name, win_team_points, lose_team_points, win_margin). Sort your results by win_margin from highest to lowest, then by win_team_points from highest to lowest.

(9) Find those teams such that each player in this team scored for this team (i.e. points $>$ 0 in at least one game) in season 2017. The output should be in format (team_name, school) and ordered by team_name alphabetically.

(10) We define a team X to be top performer in a season if no other team had more wins than X in the season. Which team or (teams) are the top performers in season 2017? The output format should be (team_name, school) and ordered alphabetically by the team name.