CS425 Database Organisation

Homework 1.1:

Design a university database that stores the following information: student, professor, and course. You are required to show the database schema (schematic view) and the data/table view of each entity.

Remember to include the metadata to self-describe the data stored in the database.

Use any professional tool for drawing (hint: Ms Visio).

Deadline: 17th Jan 2023 Time: 12:00 noon.

Homework 1.2:

Implement the database schema submitted as “Homework 1.1”. You are required to specify clearly, and completely all possible assumptions undertaken on your database design and implementation.

Use open-source RDBMS for your implementation (PostgreSQL or Workbench MySQL).

Deadline: 24th Jan 2023 Time: 12:00 noon.

Homework 1.3:

Implement the database schema (SailingDB) found in the attachment pdf below and insert the data as shown in each table. You are provided with the query statements and the SQL commands labelled from no.1 to 29, respectively. You are required to screenshot the resultant table for each SQL command.

Submit: 1) the query statement, 2) the SQL command, and 3) the screenshot showing the query result/ resultant table for each question (as a single PDF file).

Click the icon pdf below to see the homework.

PDF

SailingDB.pdf

Deadline: 31st Jan 2023 Time: 18h:00.

Homework 1.4:

Create FOUR (4) relations called Coach, Person, Player, and Team, each containing the same column names as those found in the corresponding CSV data files (see attachment on BB). Load the data into the MySQL database using the CSV files (Hint: Import the data using import wizard).

You are required to answer the following database queries (from 1 to 16) by providing the following:

- the database query,
- the SQL commands, and
- query result/resultant table (strictly screenshots).

Answer the following database queries:

1. Output the number of rows in each of your 4 relations (using 4 SELECT statements) in this order: Coach, Person, Player, Team. Call the result column LOADED each time.

2. Output everything in the Team relation, in ascending (increasing) order of TmID.

3. Show the TmID of teams with Milwaukee in their name, but your query must cater for the fact that people spell this incorrectly – everyone starts with "MIL", then somewhere later they have a "W" and even later a "K". Example misspellings are Millwaukee, Milwakee, Milwuakee, Milwaukey, etc.

4. What are the lowest and the highest number of Games (played) in the coach data? Call the first result column LOWEST and the second result column HIGHEST.

5. Show the BioID, TmID, Points and Attempts for each player that scored more than 2000 points, in decreasing order of Points. Players with the same number of Points should be shown in alphabetical order of BioID.

6. Show BioID of persons born in Gabon (GAB) and BioID of persons born in Egypt (EGY), if any.

7. Show the following for Jamesle01: his Points, the full Name of his team, & the BioID of his team's coach.

8. Show the BioID of players whose BirthCountry is in the data, but their BirthCity is not in the data.

9. Which coach(es) Won the most games? Give BioID and the number Won.

10. What percentage of players have Points scored as zero? Call the result column NONSCORERS.

11. How many teams have Lost more games than they have Won? Call the result column LOSERS.

12. How many teams belong to each ConfID? Call the 2nd column CONFSIZE.

13. How many countries do the persons in this data come from? Call the result column NUMLANDS.

14. Which pairs of teams have the exact same record (meaning the same values for Won and the same values for Lost)? Show their Won value and then their Lost value and then the 2 Names, making sure that the 3rd column Name is alphabetically before the 4th column Name so information is not repeated. Call the 3rd column TEAM1 and the 4th column TEAM2.

15. For each of the 5 Rankings in ConfID “EC”, show the average number of games Lost by those teams. Call the 2nd column AVLOSSES.

16. Give a SQL statement to output “INVALID” if any information in any row (tuple) of any relation has invalid data (e.g., the Games value is not equal to Won value plus Lost value). If all the data is valid then it should output an empty table. Call the result column ANYPROBLEMS.

Submit as a single PDF file.
Deadline: 7th Feb 2023 Time: 18h:00.