**PROJECT DESCRIPTION:**
Our CPSC 304 project will be modeling and storing a database in the domain of professional sports, encompassing: the sport being played, individuals involved in sports (i.e athletes), teams, games, leagues, seasons, sponsors, stadiums and broadcasting networks. The database will model information relevant to those working in professional sports, or fans of professional sports such as statistics of games and players, or the usage of a stadium.

Users will be able to store information in the database through the GUI by inputting the desired values into the necessary fields such as inputting the channel name and number to store a new channel. Users will be able to perform specific queries on the database through the GUI such as looking for all players born in 1998 or the team with the most winning games. Stored information will persist across different instances of the application, with the user being able to access previously stored information upon relaunching the application.

TASKS AND TIMELINE:

Back End:

- Parse string into SQL statements
    a. Insert, delete, update
    b. filtering/queries
- Storing and loading datasets
- Insert, delete, and update functions for each table
    a. Have a “Relation” interface
        - Insert, delete, update
    b. Each relation implements this interface
- Query functions
    a. Simple Select
    b. Simple filter
    c. Season schedule
    d. Broadcasters for a game
    e. All broadcaster for a league
    f. List players on a team
    g. All leagues that play a specified sport
    h. Coaches who have won a certain number of championships with the same team
- Query output array
    a. An array of comma-seperated strings, or a 2d array of strings
    b. First entry in array is attribute names

Front End:

- 2 types of users: admins who can edit data, and end users who can just query and view data.
- Front page buttons:

  - Add
  - Update
  - Remove
  - Query
- Query buttons:
  - One for each of the backend query functions
  - Have dropdown menus to limit options for user
    - First select an available relation from a list
    - Then select an attribute, etc.
- Query output:
  - Translate an array of strings into a table
- Reset/home button
  - Returns state to basic front page status

Description of Challenges:

- Making GUI user friendly
- Complex query functions related to relational algebra
- Handling 2-tiered user system
- Storing datasets to disk and reloading them
- Handling different pages for specific relations
- Handling normalised tables when inserting/updating/deleting
  - ON UPDATE CASCADE not supported
- Handling foreign keys in UI when editing data.
- Lossless join of relations, specifically normalised relations with many tables

Timeline and assignment of duties:

- Backend completion by November 27th
  - Add/remove/update/store/load data completion by November 15th
    - Hunter:
      - Storing and loading datasets
    - Aditya:
      - Add/remove data
    - David:
      - Update data
  - Queries completed by November 27th
    - Hunter:
      - Season Schedule
      - List players on a team
      - Query output
    - Aditya:
      - Simple Select

      - Coaches who have won a certain number of championships with the same team
      - All leagues that play a specified sport
    - David:
      - Simple Filter
      - Broadcasters for a game
      - All broadcaster for a league
- Frontend completion by December 1st
  - Hunter:
    - Front page: query
    - Query:
      - Season Schedule
      - List players on a team
      - Print Query output
      - Reset/home button
  - Aditya:
    - Front page: add/remove
    - Query:
      - Simple Select
      - Coaches who have won a certain number of championships with the same team
      - All leagues that play a specified sport
  - David:
    - Front page: update
    - Query:
      - Simple Filter
      - Broadcasters for a game
      - All broadcaster for a league