1. **Create tables with following constraints and then load data given in file.**

**Key Constraints**

* mID is a key for Movie,
* (title,year) is a key for Movie
* rID is a key for Reviewer
* (rID,mID,ratingDate) is a key for Rating but with null values allowed  
    
  **Non-Null Constraints**
* Reviewer.name may not be NULL
* Rating.stars may not be NULL  
    
  **Attribute-Based Check Constraints**
* Movie.year must be after 1900
* Rating.stars must be in {1,2,3,4,5}
* Rating.ratingDate must be after 2000  
    
  **Tuple-Based Check Constraints**
* "Steven Spielberg" movies must be before 1990 and "James Cameron" movies must be after 1990

Schema:

Movie ( mID, title, year, director )  
There is a movie with ID number mID, a title, a release year, and a director.  
Reviewer ( rID, name )  
The reviewer with ID number rID has a certain name.  
Rating ( rID, mID, stars, ratingDate )  
The reviewer rID gave the movie mID a number of stars rating (1-5) on a certain ratingDate.

*Copy your SQL statements onto a text file with name rollno\_sql1.txt*

1. **Run the following English queries in SQL, *copy your output onto a text file and save as rollno\_sql2.txt* and also write down their Relational Algebra Equivalents on this paper.**
2. Find the names of all reviewers who rated Gone with the Wind.
3. List movie titles and average ratings, from highest-rated to lowest-rated. If two or more movies have the same average rating, list them in alphabetical order.
4. Find the titles of all movies not reviewed by Chris Jackson.
5. For any rating where the reviewer is the same as the director of the movie, return the reviewer name, movie title, and number of stars.
6. Return all reviewer names and movie names together in a single list, alphabetized. (Sorting by the first name of the reviewer and first word in the title is fine; no need for special processing on last names or removing "The".)
7. For each rating that is the lowest (fewest stars) currently in the database, return the reviewer name, movie title, and number of stars.
8. ***For each of the following, execute and report whether it is successful or gives error. Write down the reason***
9. update Movie set mID = mID + 1;
10. insert into Movie values (109, 'Titanic', 1997, 'JC');
11. insert into Reviewer values (201, 'Ted Codd');
12. update Rating set rID = 205, mID = 104;
13. insert into Reviewer values (209, null);
14. update Rating set stars = null where rID = 208;
15. update Movie set year = year - 40;
16. update Rating set stars = stars + 1;
17. insert into Rating values (201, 101, 1, '1999-01-01');
18. insert into Movie values (109, 'Jurassic Park', 1993, 'Steven Spielberg');
19. update Movie set year = year-10 where title = 'Titanic';
20. insert into Movie values (109, 'Titanic', 2001, null);
21. update Rating set mID = 109;
22. update Movie set year = 1901 where director <> 'James Cameron';
23. update Rating set stars = stars - 1;

**Extra Credit:** For all pairs of reviewers such that both reviewers gave a rating to the same movie, return the names of both reviewers. Eliminate duplicates, don't pair reviewers with themselves, and include each pair only once. For each pair, return the names in the pair in alphabetical order.