CS-5200 Homework 7

SQL procedures, functions, triggers and prepared statements in MySQL.

This assignment gives you an opportunity to create stored procedures, functions, triggers and prepared statements from queries you created for the Lotr of the Rings schema. There is no starter file for this assignment. You should complete this assignment given the *lotrfinal* schema provided to you for homework 5.

Please submit one .sql file to canvas named LastnameFirstInitialHwk7sql where Lastname = your last name, and Firstname = you first letter of your first name. The file should contain the SQL code for each question named The .sql file should be broken into a section per question. Each section starts with a comment that lists the question number as well as the question description and any other comment you believe helps to describe the solution. This is followed by the solution. The solution is followed by test code that runs the solution, make sure you provide different executions of the solution.

1. Write a procedure track_character(character_name) that accepts a character name and returns a result set that contains a list of the other characters that the provided character has encountered. The result set should contain the character's name, the region name, the book name, and the name of the encountered character. (10 points)

2. Write a procedure track_region(region) that accepts a region name and returns a result set that contains the region name, the book name, the number of encounters for that region and the leader of that region. (10 points)

3. Write a function named strongerSpecie(sp1,sp2). It accepts 2 species and returns 1 if sp1 has a size larger than sp2, 0 if they have equal sizes, else -1 (10 points)

4. Write a function named region_most_encounters(character_name) that accepts a character name and returns the name of the region where the character has had the most encounters. (10 points)

5. Write a function named home_region_encounter(character_name) that accepts a character name and returns TRUE if the character has had a first encounter in his homeland. FALSE if the character has not had a first encounter in his homeland. or NULL if the character's homeland is not known. (10 points)

6. Write a function named encounters_in_num_region(region_name) that accepts a region's name as an argument and returns the number of encounters for that region. (10 points)

7. Write a procedure named fellowship_encounters(book) that accepts a book's name and returns the fellowship characters (all fields in the character table) having first encounters in that book. (10 points)

8. Modify the books table to contain a field called encounters_in_book and write a procedure called initialize_encounters_count(book) that accepts a book id and initializes the field to the number of encounters that occur in that book for the current encounters table. The book table modification can occur outside or inside of the procedure. (10 points)

9. Write a trigger that updates the field encounters_in_book for the book records in the lotr_book table. The field should contain the number of first encounters for that book. Call the trigger firstencounters_after_insert. Insert the following records into the database. Insert a first encounter in Rivendell between Legolas and Frodo for book 1 . Ensure that the sencounters_in_book field is properly updated for this data. (10 points)

10. Create and execute a prepared statement from the SQL workbench that calls home_region_encounter with the argument 'Aragorn'. Use a user session variable to pass the argument to the function. (5 points)

11. Create and execute a prepared statement that calls region_most_encounters() with the argument 'Aragorn'. Once again use a user session variable to pass the argument to the function. (5 points)