CSCI 620/Section 01-03/Rivero, Introduction to Big Data, Spring 2205

Assignment 4 – Document-oriented databases

Description

We will use MongoDB Community Edition 3.6 and the following collections:

Movies	**People**
{	{
_id : … ,	_id : … ,
title : … ,	name : … ,
isAdult : … ,	birthYear : … ,
year : … ,	deathYear : …
runtime : … ,	}
rating : … ,	
votes : … ,	
genres : […]	
}	
MoviesDenorm	**PeopleDenorm**
{	{
_id : … ,	_id : … ,
title : … ,	name : … ,
isAdult : … ,	birthYear : … ,
year : … ,	deathYear : … ,
runtime : … ,	moviesActed : […] ,
rating : … ,	moviesDirected : […] ,
votes : … ,	moviesProduced : […] ,
genres : […],	moviesWritten : […]
actors : […],	}
directors : […],	
producers : […],	
writers : […]	
}	

The _id fields are the ids used in the relational database. The genres fields are arrays with the names of the genres (not the ids). The rest of the arrays include person or movie ids. Empty arrays are not allowed. For instance, a movie without genres should not contain any genres field. The rest of the fields are those that translate directly from the relational database. Null fields are not allowed. For example, an alive person should not contain any deathYear field. You can find a dump of these collections in myCourses. You must use the dump for the queries.

Your tasks

1.- Provide a Gradle project named ‘IMDBSQLToMongo’ to exchange the IMDB data from MySQL to MongoDB using both JDBC and MongoDB drivers. You must create the collections specified above. Use the project template and grading software. (40 points)

2.- Provide each of the following descriptions as a single aggregation query. Check the query templates and grading software to know how to provide these queries (a document containing the initial collection and the aggregation pipeline to be executed). Note that, if you use any kind of workbench software with MongoDB, the queries may be significantly different. Your queries must work using the grading software (10 points per query)

2.1.- Recent Drama movies (released between 2015 and 2020) with a rating greater than 7 and have more than ten directors.

2.2.- Horror movies directed by people named Kathryn with runtime less than 100. (Hint: Use a regular expression as follows: /^Kathryn.*/)

2.3.- Alive writers that have written more than five Sci-Fi movies between 1970 and 1990.

2.4.- Ratings of Comedy movies with more than 10,000 votes and produced by more than four alive producers. (Hint: Use "{ $divide : [{ $trunc : { $add : [{ $multiply : ['$X', 100] }, 0.5] } }, 100] }" to round the X field[1].)

2.5.- Actors that have acted in more than seven Sci-Fi movies grouped by birth and death years, i.e., if two of these actors have the same birth and death years, they should appear together.

2.6.- Alive actors that have acted in more than 25 Comedy movies and have directed at least two Action movies.

Submission instructions

- Use the software template provided in myCourses.
- Submit a single ZIP file to myCourses that must be named as your RIT user, e.g., crrvcs.zip. Do not include '@rit.edu.' The file must contain a folder named 'IMDBSQLToMongo' containing your Gradle project, and a folder named 'Queries' containing your MongoDB queries.
- Everything will be graded on a Linux machine, so you must always use the exact names provided in this document, software template and grading software.

Grading rubric

- Check the grading software.

[1] https://stackoverflow.com/questions/17482623/rounding-to-2-decimal-places-using-mongodb-aggregation-framework