

The project is two parts, MapReduce and MongoDB

- For the two parts you need to discuss them, no submission on Moodle.
- For the MapReduce part you can write code on any IDE, and you must run it on the VM.
 - If it not on the VM you loose two marks, but still your code has to be in Java project that has all dependencies.
- Regarding the MongoDB, prepare the queries, and in the discussion show how to run them on MongoDB.
 - If not you loose 2 marks
- Last allowed date of submission is 18/5/2023,

Part 1: MapReduce

Create and samples of the following relations, each relation represents a dataset of text files stores on HDFS.

1. ratings (UserID, MovieID, Rating) // where rating represent the rating between (from 1 to 5) given by the user to the corresponding movieID
2. users (UserID, Gender, Age)
3. movies (MovieID, Title, Genres) // where genres in the classification of the movie such as comedy, children, action,

Suppose you have been given a task to find the average rating for each movie in the form (movieID, Title, avg_rating). Computing the average rating must consider the following:

4. only children and comedy movies
 5. consider rating values that are above 2
 6. consider ratings from users who's age is above 25
- Create a java project that contains MapReduce code to achieve the above described task, write as much as needed MapReduce jobs.

Part 2: MongoDB

For each of the following, you need to write a query and also show the output (result returned from the shell)

1. Create a new database called gamesDB
2. Write a query to make sure that you are using the gamesDB
3. Create a new collection called games, make sure it has been created
4. Write query to make sure that the collection was created
5. Add 5 games to the games collection; give each one of them has the following properties: name, publisher, year_released, and rating (value from 1 to 5)
6. write a query to return all games in the collection
7. write a query that return only 3 games
8. write a query to return the top 3 games based on rating value
9. write a query that return games whose rating is 5 and released after 2007
10. update the game whose rating is 3 to be 4