Data Science and Database Technologies

Homework 4 – MongoDB

Introduction

The “bike stations” dataset is available, containing information about 65 stations of a bike sharing service. To carry out the homework, it is recommended to import the dataset as a collection on a database on a local MongoDB server, using the `mongoimport` tool.

Note: To use locally MongoDB, install the “MongoDB Community Edition” and refer to the official documentation.

Note: To use the `mongoimport` tool, install “MongoDB Database Tools”, (MongoDB Database Tools).

You can refer to the tutorial and material provided in the MongoDB lab for the commands for importing a collection, running a MongoDB server locally and accessing the MongoDB command-line interface (shell).

An example of a station extracted from the collection is reported below.


```
{
        "_id" : ObjectId("61b75b13fd4d2d1ea82e75f4"),
        "empty_slots" : 10,
        "extra" : {
                "number" : 57,
                "reviews" : 222,
                "score" : 4,
                "status" : "online",
                "uid" : "307"
        },
        "free_bikes" : 4,
        "id" : "bfa12cb895ac0d7392dde60b6b433cdf",
        "name" : "San Francesco da Paola",
        "timestamp" : "2021-12-10T14:54:39.185000Z",
        "location" : {
                "type" : "Point",
                "coordinates" : [
                        45.068617,
                        7.689097
                ]
        }
}
```

To answer the homework questions, it is necessary to report:
- The query used to obtain the answer to the question (the query must extract only the fields necessary to answer the question)
- The result of the question

Question 1

How many stations have (`extra.status`) “online”status. How many stations have “offline” status?

Question 2

How many stations have a status different than “online” e “offline”?

Question 3

For stations that have a status different than “offline” and “online” status, visualize only the value of the status field.

Question 4

What are the active stations (status = online) with an average rating (`extra.score`) greater than or equal to 4?

Extract the list of the names of these stations, sorted in alphabetical order.

Question 5

What is the name of the inactive stations (status = offline) that have at least one free slot (empty_slots> 0) or have at least one bike available (free_bikes> 0)? How many free slots and how many bikes are available?

Question 6

What is the total number of reviews (extra.reviews) for all stations?

Question 7

For each value of average ratings (score), how many stations have that rating? Sort the result by descending rating.

Question 8

What is the average rating for active (status = online) and inactive (status = offline) stations?

Note: Stations that do not fit into either category (see question 3) will not be considered in the count query.

Question 9

What are the average ratings for stations without bikes (free_bikes = 0) and for those with at least one bike available (free_bikes> 0)?

Hint: You can use the map-reduce to answer this question. The mapReduce () function was deprecated in MongoDB 5.0. However, the paradigm remains a viable alternative, used, for example, in Hadoop. For this reason, its use is recommended for the resolution of this exercise.

Question 10

Answer question 9, referring only to active stations (status = online).

Hint: Also for this exercise, the use of the map-reduce paradigm is recommended.

Question 11

What are the names of the 3 stations with available bikes (free_bikes> 0) closest to the point [45.07456, 7.69463]? How many bikes are available?

Note: You need to create a 2dsphere index on "location" to use the $near operator.

Note: You can use the `limit(n)` method to limit the number of results extracted.

Question 12

What are the names of the 3 stations with available bikes (free_bikes> 0) closest to the "Politecnico 4" station? How many bikes are available?

Note: You need to create a 2dsphere index on "location" to use the $near operator.

Requirement: Solve the exercise using a nested query to extract the position of the "Politecnico 4" station.