Big Data Technologies

2024/2025

Lab 5

This lab proposes you to test several NoSQL databases.

- **First steps in MongoDB**

MongoDB is a document-oriented NoSQL database used for high volume data storage.
To start the MongoDB service, run Docker Desktop, then in the terminal, use the commands

```
docker run -d --name mongo-demo \
-e MONGO_ROOT_USERNAME=mongoadmin \
-e MONGO_ROOT_PASSWORD=LikeAndSubscribe \
-p 27017:27017 \
mongo:7.0

docker exec -it mongo-demo mongosh
```

From this way, we are using the official MongoDB image.

Answer the following questions.

1. In MongoDB, databases hold collections of documents. To select a database to use, in the mongo shell, run the command “db”. You can switch to a non-existent database with the following command “use myNewDB” where “myNewDB” is the name of the new database. Switch to the database “myDB”.
2. To create a collection associated to the current database, execute the command:

   db.createCollection('firstCollection');

3. Insert the first JSON document into the “firstCollection” collection created above.

   db.firstCollection.insertOne({name:'John', age: 26, skill:'MongoDB'});

   For each inserted record, the field “_id” is automatically added. The field “_id” is reserved for use as a primary key; its value must be unique in the collection, is immutable, and may be of any type other than an array.

4. The insertion can be verified by using the command shown below:

   db.firstCollection.find();

   You can also use the command “show collections” to show all the collections of the database.

5. Insert a second JSON document with the information: “_id=19”, “name=Peter”, “age=28”, “skill=HBase” and “company=TheBigData”. Is the second JSON document consistent with the first one?

6. Is it possible to insert two times the same JSON document (without the same "_id")?
7. Insert the following JSON documents into the "firstCollection" collection:

name	age	skill	company	country
Edward	31	Java, MongoDB, Python		USA
Mary	24	Python	TheBigData	Italy
Julia	25	Java, HBase, Python	SaveTheWorld	France

Hints: you can use the command "db.firstCollection.insertMany" to populate the "firstCollection" collection with several documents in one call.

8. To execute a MongoDB equivalent of a SQL like clause, MongoDB uses regex. Regex is a series of characters forming a pattern to match.

   What is the meaning of the query: db.firstCollection.find({skill:/.*Java.*/});
9. What is the meaning of the query: db.firstCollection.find({age:{$lt:26}});
10. Create a query that selects all the workers whose skill contains "Java" and age is strictly lower than 26. Hints: explanations to write a query are given on the page

    https://docs.mongodb.com/manual/reference/method/db.collection.find
11. Use the command "deleteOne" to remove the document with "_id" equal to 19.
12. Delete all the documents of the collection.
13. Remove the "firstCollection" collection.
14. Quit the mongo shell with "quit()".

- **First steps in Cassandra [This exercise is optional. Do it at home.]**

Cassandra is a NoSQL database which is distributed and scalable. It is provided by Apache. To start Cassandra, run Docker Desktop, then in the terminal, use the commands

```
docker run --rm -d --name cassandra --hostname cassandra cassandra

docker exec -it cassandra cqlsh
```

Warning: sometimes you must wait a while for the machine to fully boot after "docker run" before executing the command "docker exec". If you execute "docker exec" too soon, you will have an error message. Just wait!

For your information, the command "`cqlsh`" starts the Cassandra shell. The following page gives more information on the Cypher shell and its syntax:

https://cassandra.apache.org/doc/4.1/cassandra/cql/index.html

To exit the Cassandra shell, just write "`exit`".

Answer the following questions.

1. A keyspace is an object that is used to hold column families, user defined types. Create a keyspace named "bigdata" with a simple strategy and a replication factor equal to 1.
2. Run the command "DESCRIBE keyspaces;" to check if your keyspace exists.
3. To use the created keyspace, you have to use the "USE" command.
4. Create a table named "student" with the columns "student_id" (integer), "student_name" (text), "student_city" (text), "student_fees" (integer) and "student_phone" (integer). The primary key is "student_id".
5. We need to insert some data in student table. Insert the following student with the command

   INSERT INTO student (student_id, student_fees, student_name) VALUES(1,5000, 'Alice');

student_id	student_fees	student_name
1	5000	Alice
2	3000	Bob
3	2000	Christopher

6. You can use SELECT command to verify whether data is inserted or not.
7. What are the values of the missing columns?
8. Create a query to get the name of the student whose id is 2.
9. Update the feeds of the student whose id is 3. The new value of the fees is 2500.
10. Delete the student_fees where student_id is 1.
11. Alter the table "student": add a new column "student_email" (text).
12. Drop the table "student".
13. Use the command "DESCRIBE tables;" to check that the table is removed.
14. Drop the keyspace "bigdata"

- **First steps in Redis [This exercise is optional. Do it at home.]**

Redis is a NoSQL data structures server based on a key-value storage. It is a data structures server, supporting different kinds of values. What this means is that in Redis the value is not limited to a simple string, but can also hold more complex data structures.

To start both the Redis server and Redis client, run Docker Desktop, then in the terminal, use the commands

```
docker run -d --name redis-demo -p 6379:6379 redis

docker exec -it redis-demo redis-cli
```

From this way, we are using the official Redis image.
You can use the client shell to manage the data in Redis.

Answer the following questions.

1. An introduction to the Redis commands is given in
   https://redis.io/topics/data-types-intro
   Create the key-value couple (“mykey”, “myvalue”)
2. Get the value associated to the key “mykey”
3. If you insert a new value with the same key, what is the result?
4. Insert the three following key-value couples in a single command: (k1, firstValue), (k2, secondValue) and (k3, thirdValue)
5. Get the two values associated to the keys k1 and k2 in a single command.
6. List all the keys stored in the database.
7. Delete the keys k1 and k3.
8. Store in the database the half-byte “1011” as a bitmap.
9. Quit the client with “quit”.

NB: you can use Redis from Spark. The insertion of complex key-value couple is easier, especially when the value is a bitmap (image, etc.).

- **First steps in Neo4j [This exercise is optional. Do it at home.]**

Neo4j is a graph database management system. To start Neo4j, run Docker Desktop, then in the terminal, use the commands

```
docker run --publish=7474:7474 --publish=7687:7687 --env NEO4J_AUTH=none --name neo4j-demo -d neo4j

docker exec -it neo4j-demo cypher-shell
```

To exit the Cypher shell, just write ":exit".

The following page gives more information on the Cypher shell and its syntax:
https://neo4j.com/developer/cypher/syntax/

Answer the following questions.

1. The node is surrounded with parentheses in Cypher, e.g. (node). If we later want to refer to the node, we can give it a variable like (p). You can assign labels to a node. Labels are kind of like tags and allow you to specify certain types of entities to look for or create. Create a node with variable "bigdata" and label "Topic".
2. You can assign one (or more) property to a node. A Property is a key-value pair. Create a node with variable "michael", label "Person" and properties "name: 'Michael'" and "title: 'Expert'".
3. Create a note with variable "Jessica" and label "Person".
4. We can create a relationship between two nodes, Create the relationship with variable "rel" and type "LIKES" between "michael" and "bigdata". A relationship type is similar to a node label.
5. Create the relationship with label "KNOWS" between "michael" and "jessica".
6. Create with a single command the two nodes "(jenn:Person {name: 'Jennifer'})" and (tbc:Company {name: 'TheBigCompany'}) with the relationship type "WORKS_FOR". The sentence with the two nodes and the relationship is called a pattern.
7. Find all the nodes. Hints: use the keywords "MATCH" and "RETURN".
8. Find the labeled "Person" nodes in the graph.
9. Find Person nodes in the graph that have a title of 'Expert'.
10. Find which Company Jennifer works for.
11. Find the type of relationship between "michael" and "bigdata".
12. Delete all nodes and relationships.