BUAN 6320

Database Foundations for Business Analytics

Project 3

Project 3 includes 2 parts, the first part deals with indexing and query timing where you will try to evaluate the impact of indexes on your database, and the second part deals with MongoDB and MQL where you will try to create a database from a JSON file using MongoDB.

- Please document all your steps in the project report
- Project reports should not exceed 10 pages

Part 1: Indexing and Query Timing

In this part you will have to work with the database you created from the previous projects and try indexing your database. Your database currently has only indexes automatically created by MySQL (your database management system). The goal of this step is to manually create indexes on columns and evaluate the performance of the queries before and after adding these indexes by timing them.

1.1. List all the current indexes in your database and the columns they are associated with along with the index type
1.2. Explain what is in common between these columns (why these columns are indexed automatically by the database management system)
1.3. Make a copy of your database and delete all the indexes there (you might need to delete foreign keys before you can delete some of the indexes) – now you have two databases: database A with indexes and database B without any indexes.
1.4. Write at least 5 queries (with JOINs between your tables)
1.5. Execute and time these queries on both databases and report your findings (repeat timing for each query at least 10 times and average the times)
1.6. Select some columns from database A (columns that are not already indexed) and create index on them
1.7. Write a query for each column – the query should include the column in the WHERE clause in a condition
1.8. Execute and time these queries on both databases and report your findings (repeat timing for each query at least 10 times and average the times)
1.9. Make a conclusion based on your findings in this part

Part 2: MongoDB and MQL

For this part of the project, you need to use one of the datasets provided below.

1. Pakistan's Largest PakWheels Automobiles Listings
2. Bikes from Bikez.com

(The raw files from these two datasets on Kaggle requires some modifications before importing into MongoDB – The modified files are uploaded in eLearning for your ease)

After you have selected your dataset:

2.1. Explore your dataset and familiarize yourself with the dataset and its content
2.2. Explain why it is better to use non-relational databases such as MongoDB to work with such a dataset (explain in the context of your dataset)

2.3. Import your dataset into MongoDB using either MongoDB Compass or MongoDB Database Tools*

2.4. List some of the attributes (field/properties) of your database which are common among all documents.

2.4.1. For these fields, provide some of the values they contain in the database

2.5. List some of the attributes (field/properties) of your database which are not common among all the documents

2.5.1. For these fields, provide some of the values they contain in the database

2.6. Write at least 5 queries using the key-value pairs you found in the previous steps to narrow down the result (provide the queries and results in your report)

2.7. Write at least 5 update queries to update some of the values in your database (provide the queries and results in your report)

2.8. Write at least 5 queries to insert new documents into your database (provide the queries and results in your report)

2.9. Write at least 5 delete queries to remove documents from your database (provide the queries and results in your report)

---

* Importing a dataset using MongoDB Database Tools (instructions here):

1. Download MongoDB Database Tools from here.
2. Extract the files and copy the contents of bin folder to the bin folder of the MongoDB installed on your computer
3. Add the path of this folder to your system's path so hat you can easily run these commands in the system's terminal (instructions for windows)
4. In your system's terminal, execute the following command:

```
mongoimport --db <db> --collection <coll> --file <path> --jsonArray
```

where `<db>` is the name of the database you want to import the data into, `<coll>` is the name of the collection you want to import the data into and `<path>` is the path to the file.