MongoDB Exercises for Dec 2020 batch

EXERCISE

(1) Create Database

1-1. Create a database called **mydatabase**, drop it and create it again. Check which database you are currently in.

Create a Collection & Insert a Record

1-2. Create a collection called customers in **mydatabase** created in Exercise 1 and insert the document below. Check if the document is inserted correctly.

```
{ "firstName":"John",
  "lastName":"West",
  "email":"john.west@mail.com",
  "phone":"032345432134",
  "BusinessType": ["Sell", "Sugar", "Drinks"],
  "Reference":100,
  "Company":"Coca-Cola"}
```

(2) Bulk Load JSON File

2-1. Create a collection called **transactions** in **mydatabase** (drop if it already exists) and bulk load the data from a json file, **transactions.json** (see the data at the end of the questions).

2-2. Append the records with the same file, **transactions.json**

2-3. Upsert the record from the new file called transactions_upsert.json (see the data at the end of the questions

(3) Query MongoDB with Conditions

This question uses the collection (transactions) created in Exercise 3.

3-1. Find any record where Name is **Tom**

3-2. Find any record where total payment amount (**Payment.Total**) is 400.

3-3. Find any record where price (**Transaction.price**) is greater than 400.

3-4. Find any record where Note is null or the key itself is missing.

3-5. Find any record where Note exists and its value is null.

3-6. Find any record where the Note key does not exist.

4) CRUD Operations

This question uses the collection (transactions) created in Exercise 3. CRUD: Create, Read, Update and Delete.

4-1. Insert a record below.

4-2. Updating the new inserted record above. Make Name='Updated Record' & Note='Updated!'

4-3. Delete the record inserted above by using Id.

(5) Aggregation with MongoDB

This question uses the collection (transactions) created in Exercise 2.

5-1. Calculate the total transaction amount by adding up Payment.Total in all records.

5-2. Get the total price per record by adding up the price values in the Transaction array (Transaction.price).

5-3. Calculate total payments (Payment.Total) for each payment type (Payment.Type).

5-4. Find the max Id.

5-5. Find the max price (Transaction.price).

.