**Mongo Lab2**

**(1) Create Database**

1-1. Create a database called usermanaged. Check which database you are currently in.

**(2) Create a Collection & Insert a Record**

1-2. Create a collection called customers in usermanaged 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"}

**(3) Bulk Load JSON File**

3-1. Create a collection called transactions in usermanaged ,bulk load the data from a json file(import Mongo\_EX3.1.json).

3-3. Upsert the record from the new (import Mongo\_EX3.3.json)

**(4) Bulk Load CSV File**

4-1. Create a collection and load data from a CSV file will multiple rows. Define the keys from the header row.

**(5) Query MongoDB with Conditions**

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

5-1. Find any record where Name is Tom

5-2. Find any record where total payment amount (Payment.Total) is 400.

5-3. Find any record where price (Transaction.price) is greater than 400.

5-4. Find any record where Note is null or the key itself is missing.

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

5-6. Find any record where the Note key does not exist.

**(6) Aggregation with MongoDB**

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

6-1. Calculate the total transaction amount by adding up Payment.Total in all records.

6-2. Get the total price per record by adding up the price values in the Transaction array (Transaction.price).

6-3. Calculate total payments (Payment.Total) for each payment type (Payment.Type).

6-4. Find the max Id.

6-5. Find the max price (Transaction.price).

**(7) CRUD Operations**

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

7-1. Insert a record below.     {  
    "Id": 110,  
    "Name": "Inserted Record",  
    "TransactionId": "tranNew1",  
  "Transaction": [  
    {  
    "ItemId":"c324",  
    "price": 456  
    },  
    {  
    "ItemId":"d456",  
    "price": 543    
    }  
  ],  
  "Subscriber": false,  
  "Payment": {  
    "Type": "Debit-Card",  
    "Total": 999,  
    "Success": true  
  },  
  "Note":'Hello World'  
})

7-2. Updating the new inserted record above. Make Name=’Updated Record’ & Note=’Updated!’

7-3. Delete the record inserted above by using Id.

**(8) User Creation**

8-1. Create a read only user who can query records from collections from all databases.

8-2. Create a writer user who can create collections and do CRUD operations in any collections.

8-3. Create a usermanaged user who can do the writer operation in the usermanaged database and read only for the rest of the databases.