REPORT

I Have completed my SQL lab with help of notes provided during the regular online classes as well as reference book shared in class that is "book **BeginningSQLqueries** by Clare Churcher".

I got required the output (correct format) after rectifying silly error in almost each queries. I Refer website such as https://stackoverflow.com/, https://www.geeksforgeeks.org/ https://www.w3schools.com/ , to get know different ways of writing queies.

Department of Data Science, Bishop Heber College Tiruchirappalli
NoSQL Database Management Lab

Lab11. Ecommerce Product Catalog Design using MongoDB

In order to manage an ecommerce system, the first thing you need is a product catalog. Product catalogs must have the capacity to store many different types of objects with different sets of attributes. These kinds of data collections work quite well with MongoDB's flexible data model, making MongoDB a natural fit for this type of data.

Question1. Create a collection **Product** and insert the following document for Audio Album

At the beginning of the document, a document-based schema should contain general product information, to facilitate searches of the entire catalog. After the common fields, we'll add a details subdocument that contains fields that vary between product types such as Audio Album or Movies.

```
{
        _id: "p01",
        type: "Audio Album",
        title: "Bergmans volume15",
        description: "by Fr Bergmans",
        release_date:"2020-02-15"

        shipping: {
            weight: 6,
            dimensions: {
                width: 10, height: 10, depth: 1  },
        },

        pricing: {
                list: 1200, retail: 1100, savings: 100, percentage_savings: 8
        },

        details: {
                title: "Jeba Thotta Jeya Geethangal",
                artist: "Fr Bergmans",
                genre: [ "folk", "Spiritual" ],

                tracks: [
                        "Song1", "Song2, "Song3", Song4"
                ],
        },
}
```

Question2. Insert the following document for a Movie to **Product** collection

A movie item would have the same fields for general product information, shipping, and pricing, but different fields for the **details** subdocument.

```
{
        _id: "p02",
        type: "Movie",
        title: "Master",
        description: "Movie by Actor Vijay",
        release_date:"2020-03-14"


        shipping: {
            weight: 10,
            dimensions: {
                width: 20, height: 10, depth: 2  },
        },

        pricing: {
                list: 15000, retail: 14000, savings: 500, percentage_savings: 12
        },

        details: {
                title: "Master",
                director: [ "Rex", "Peter" ],
                writer: [ "Sam", "Jones" ],
                length: "3 Hours"
        },
}
```

Question3. Show all documents in the collection Product

```
> db. Products.  find(). pretty ()
```

Question4. all products with a discount greater than 10%, sorted by descending percentage discount

```
>db. products - find ({ "pricing. list": { $gt: 10}}). sort({"pricing.
   percentage_savings": 1});
```

Question5. Find documents for the audio albums of folk genre, sorted in reverse chronological order

> db. products. find ({"details. genre": "folk"}) .sort ({_id:-1})

Question6. Find films that is acted by actor 'Vijay', sorted by release date

> db. products. Find ({ description: "Movie by Actor vijay"},
   { title:1, -id:0}).

Question7. Find movies with a particular word in the title

> db. product. Find({ title: "Master"})

Question8. Write atleast 3 queries for Product collection that are useful for Ecommerce website

1) > db. products. find ({ }, { Pricing:1, type:1, -id:0})

2) > db. products. find ({"shipping..weight": { $ not: { $ gte:9}}},
      { pricing:1, type:1, -id:0})

```
3) > db.products.find({"pricing.savings": {$gte: 300}},
   {pricing:1, type:1, details:1, _id:0})
```

LAB 11: ECOMMERCE PRODUCT CATALOG DESIGN USING MONGODB

QUSETION 1: Create a collection PRODUCT and insert the following documents for Audio Album

>db.products.insert({_id:"p01", type: "Audio Album", title:"Bergmans volume15", description:"by Fr Bergmans", release_date:"2020-02-15", shipping:{weight:6, dimensions:{width:10, height:10, depth:1}}, pricing:{list:1200,retail:1100, savings:100, percentage_savings:8}, details:{title:"jeba thotta jeya geethangal", artisrt:"Fr Bergmans", genre:["folk","Spiritual"],tracks:["Song1","Song2","Song3","Song4"],},},)

WriteResult({"nInserted":1})

QUESTION 2: insert the following documents for a movie to PRODUCT collection

>db.products.insert({_id:"p02", type:"Movie", title:"Mater", description: "Movie by Actor Vijay", release_date:"2020-03-14", shipping:{weight:10, dimensions:{width:20, height:10, depth:2}}, pricing:{list:15000, retail:14000, savings:500, percentage_savings:12}, details:{title:"Master", director:["rex","peter"], writer:["sam", "jones"], length:"3 hours"},})

WriteResult({ "nInserted" : 1 })

QUESTION 3: show all documents in the collection PRODUCT

> db.products.find().pretty()

{

    "_id" : "p01",

    "type" : "Audio Album",

    "title" : "Bergmans volume15",

    "description" : "by Fr Bergmans",

    "release_date" : "2020-02-15",

    "shipping" : {

        "weight" : 6,

        "dimensions" : {

            "width" : 10,

```json
            "height" : 10,
            "depth" : 1
        }
    },
    "pricing" : {
        "list" : 1200,
        "retail" : 1100,
        "savings" : 100,
        "percentage_savings" : 8
    },
    "details" : {
        "title" : "jeba thotta jeya geethangal",
        "artist" : "Fr Bergmans",
        "genre" : [
            "folk",
            "spiritual"
        ],
        "tracks" : [
            "song1",
            "song2",
            "song3",
            "song4"
        ]
    }
}
{
    "_id" : "p02",
    "type" : "Movie",
    "title" : "Mater",
```

```json
        "description" : "Movie by Actor Vijay",
        "release_date" : "2020-03-14",
        "shipping" : {
            "weight" : 10,
            "dimensions" : {
                "width" : 20,
                "height" : 10,
                "depth" : 2
            }
        },
        "pricing" : {
            "list" : 15000,
            "retail" : 14000,
            "savings" : 500,
            "percentage_savings" : 12
        },
        "details" : {
            "title" : "Master",
            "director" : [
                "rex",
                "peter"
            ],
            "writer" : [
                "sam",
                "jones"
            ],
            "length" : "3 hours"
        }
}
```

QUESTION 4: All products with a discount greater than 10%, sorted by descending percentage discount.

> db.products.find({"pricing.list":{$gt:10}}).sort({"pricing.percentage_savings":1});

{ "_id" : "p01", "type" : "Audio Album", "title" : "Bergmans volume15", "description" : "by Fr Bergmans", "release_date" : "2020-02-15", "shipping" : { "weight" : 6, "dimensions" : { "width" : 10, "height" : 10, "depth" : 1 } }, "pricing" : { "list" : 1200, "retail" : 1100, "savings" : 100, "percentage_savings" : 8 }, "details" : { "title" : "jeba thotta jeya geethangal", "artisrt" : "Fr Bergmans", "genre" : ["folk", "Spiritual"], "tracks" : ["Song1", "Song2", "Song3", "Song4"] } }

{ "_id" : "p02", "type" : "Movie", "title" : "Mater", "description" : "Movie by Actor Vijay", "release_date" : "2020-03-14", "shipping" : { "weight" : 10, "dimensions" : { "width" : 20, "height" : 10, "depth" : 2 } }, "pricing" : { "list" : 15000, "retail" : 14000, "savings" : 500, "percentage_savings" : 12 }, "details" : { "title" : "Master", "director" : ["rex", "peter"], "writer" : ["sam", "jones"], "length" : "3 hours" } }

QUESTION 5: Find the documents for the audio album of folk genre,sorted in reverse choronological order.

> db.products.find({"details.genre":"folk"}).sort({_id:-1})

{ "_id" : "p01", "type" : "Audio Album", "title" : "Bergmans volume15", "description" : "by Fr Bergmans", "release_date" : "2020-02-15", "shipping" : { "weight" : 6, "dimensions" : { "width" : 10, "height" : 10, "depth" : 1 } }, "pricing" : { "list" : 1200, "retail" : 1100, "savings" : 100, "percentage_savings" : 8 }, "details" : { "title" : "jeba thotta jeya geethangal", "artist" : "Fr Bergmans", "genre" : ["folk", "spiritual"], "tracks" : ["song1", "song2", "song3", "song4"] } }

QUESTION 6: Find films that is acted by actor 'vijay' , sorted by release date

> db.products.find({description:"Movie by Actor Vijay"}, {title:1,_id:0}) . sort({release_date:1})

{ "title" : "Mater" }

QUESTION 7: Find movies with a particular word in the title.

> db.product.find({title:"Master"});

{ "_id" : "p02", "type" : "Movie", "title" : "Master", "description" : "Movie by Actor Vijay", "release_date" : "2020-03-14", "shipping" : { "weight" : 10, "dimensions" : { "width" : 20, "height" : 10, "depth" : 2 } }, "pricing" : { "list" : 15000, "retail" : 14000, "savings" : 500, "percentage_savings" : 12 }, "details" : { "title" : "Master", "director" : ["Rex", "Peter"], "writer" : ["Sam", "Jones"], "length" : "3 Hours" } }

QUESTION 8: Write atleast 3 queries for product collection that are useful for Ecommerce website.

(1) >db.products.find({},{pricing:1,type:1,_id:0})

{ "type" : "Audio Album", "pricing" : { "list" : 1200, "retail" : 1100, "savings" : 100, "percentage_savings" : 8 } }

{ "type" : "Movie", "pricing" : { "list" : 15000, "retail" : 14000, "savings" : 500, "percentage_savings" : 12 } }

(2) >db.products.find({"shipping.weight":{$not:{$gte:9}}},{pricing:1,type:1 ,_id:0})

{ "type" : "Audio Album", "pricing" : { "list" : 1200, "retail" : 1100, "savings" : 100, "percentage_savings" : 8 } }

(3) >db.products.find({"pricing.savings":{$gte:300}},{pricing:1,type:1, details:1,_id:0})

{ "type" : "Movie", "pricing" : { "list" : 15000, "retail" : 14000, "savings" : 500, "percentage_savings" : 12 }, "details" : { "title" : "Master", "director" : ["rex", "peter"], "writer" : ["sam", "jones"], "length" : "3 hours" } }

REPORT

I Have completed my SQL lab with help of notes provided during the regular online classes as well as reference book shared in class that is "book **BeginningSQLqueries** by Clare Churcher".

I got required the output (correct format) after rectifying silly error in almost each queries. I Refer website such as https://stackoverflow.com/, https://www.geeksforgeeks.org/ https://www.w3schools.com/ , to get know different ways of writing queies.

Lab12. Neo4J Play Ground Exercise

In this lab, you will listen to videos of Introduction to Neo4J (www.neo4j.com) online course by using their sample database named Movies Graph. Also, you will complete the exercises of each chapter and acquire free certificate.

Neo4j GraphAcademy: Certificate of Completion

Presented to:

Maheshvaran S

Successfully completed the course: *Overview of Neo4j 4.x*

Certificate Number: 35101920

Date: 2021-2-11

The certificate holder has successfully completed the Neo4j GraphAcademy course named above.

Emil Eifrem

Emil Eifrem
CEO, Neo4j, Inc.

Neo4j GraphAcademy: Certificate of Completion

Presented to:

Maheshvaran S

Successfully completed the course: *Querying with Cypher in Neo4j 4.x*

Certificate Number: 32502078

Date: 2021-2-11

The certificate holder has successfully completed the Neo4j GraphAcademy course named above.

Emil Eifrem

Emil Eifrem
CEO, Neo4j, Inc.

Neo4j GraphAcademy: Certificate of Completion

Presented to:

Maheshvaran S

Successfully completed the course: *Creating Nodes and Relationships in Neo4j 4.x*

Certificate Number: 70518904

Date: 2021-2-11

The certificate holder has successfully completed the Neo4j GraphAcademy course named above.

Emil Eifrem

Emil Eifrem
CEO, Neo4j, Inc.

Neo4j GraphAcademy: Certificate of Completion

Presented to:

Maheshvaran S

Successfully completed the course: *Using Indexes and Query Best Practices in Neo4j 4.x*

Certificate Number: 86100635

Date: 2021-2-12

The certificate holder has successfully completed the Neo4j GraphAcademy course named above.

Emil Eifrem

Emil Eifrem
CEO, Neo4j, Inc.

Neo4j GraphAcademy: Certificate of Completion

Presented to:

Maheshvaran S

Successfully completed the course: *Importing Data with Neo4j 4.x*

Certificate Number: 639989353

Date: 2021-2-12

The certificate holder has successfully completed the Neo4j GraphAcademy course named above.

Emil Eifrem

Emil Eifrem
CEO, Neo4j, Inc.