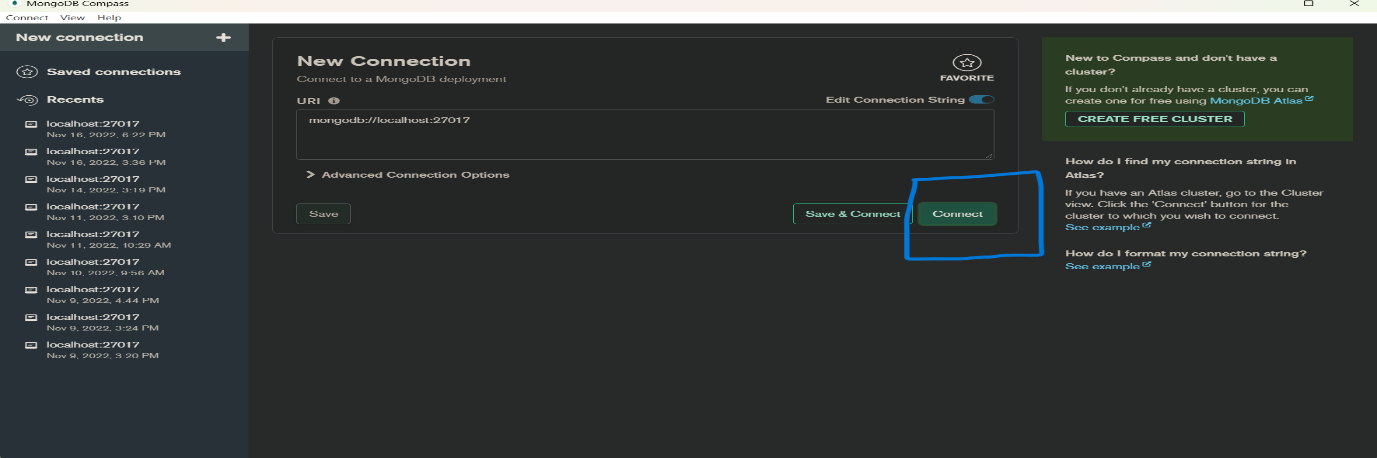
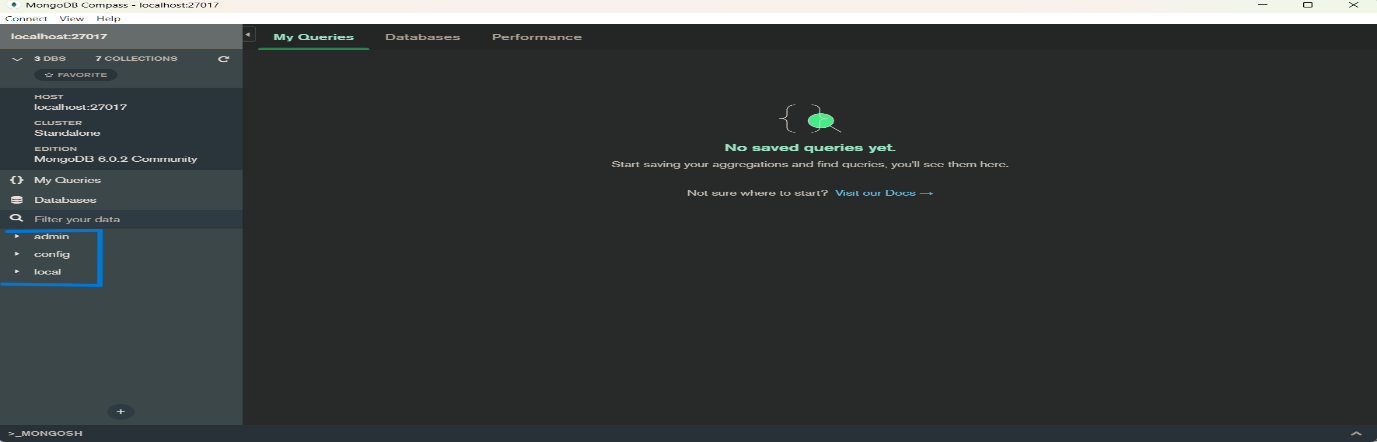
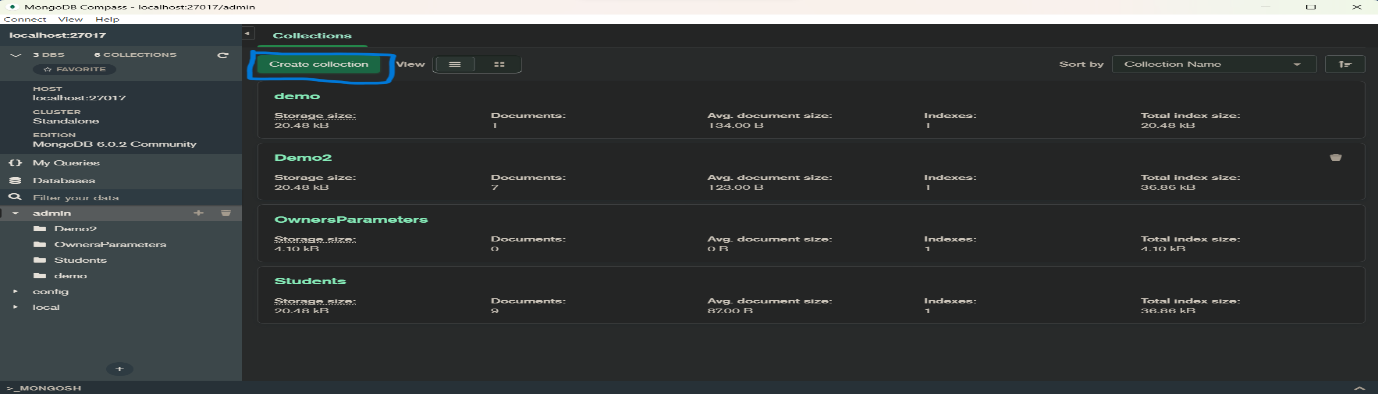
[Topics] = [Required Software/Tools], [Steps], [Reference Documentation]

[Required Tools]

1.MongoDB Shell-: <https://www.mongodb.com/try/download/shell>

2.MongoDB Compass-: <https://www.mongodb.com/try/download/shell>

1. Open MongoDB compass and click on Connect.  
   
2. There are 3 existing Database  
   a.admin  
   b.config  
   c.Local  
   
3. In existing Database, you can choose anyone one for creating collection
4. Now click on admin database.
5. Click on Create Collection and create a collection.  
   
6. Now are collection is Created.
7. Click on that collection.
8. Now Click on Add Data button.
9. In that box you can add multiple and single data and then click on Insert button.  
   ex. [

{ "item": "journal", "qty": 25, "size": { "h": 14, "w": 21, "uom": "cm" }, "status": "A" },

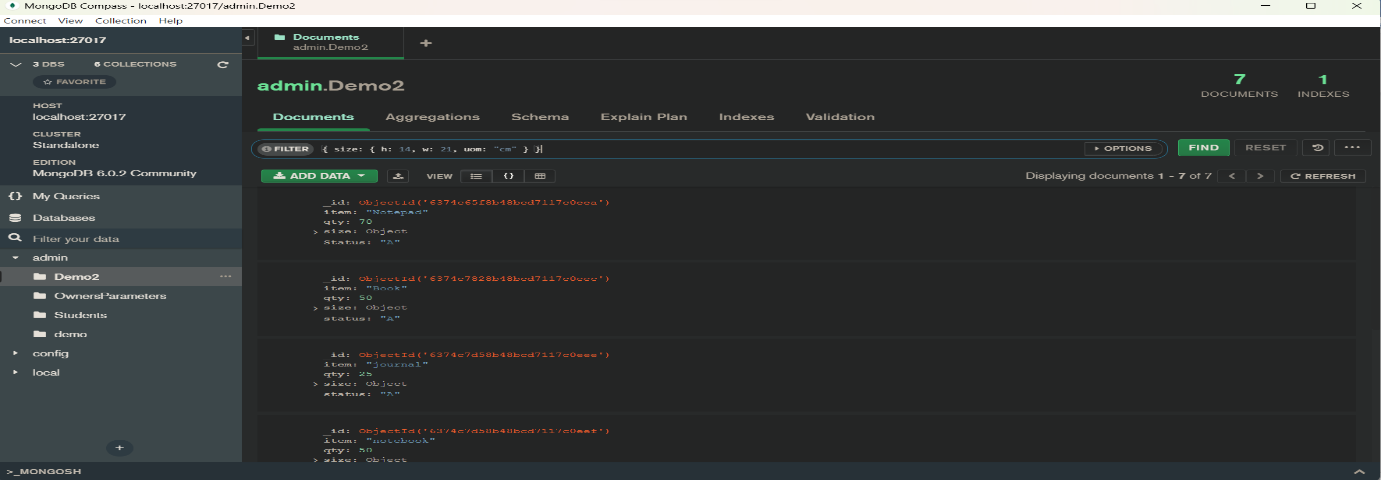
{ "item": "notebook", "qty": 50, "size": { "h": 8.5, "w": 11, "uom": "in" }, "status": "A" },

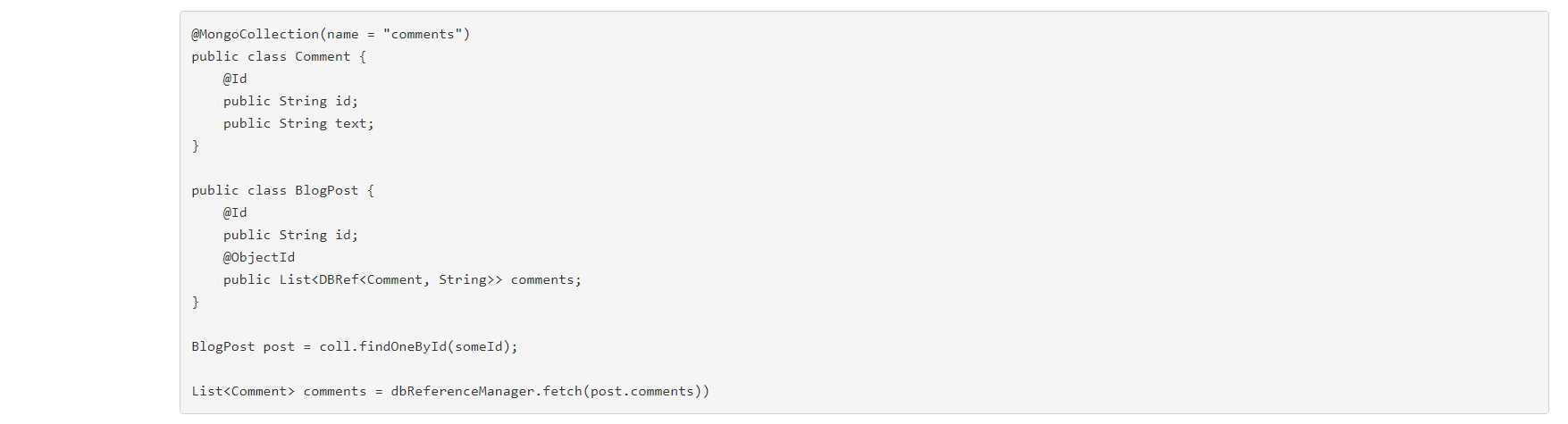
{ "item": "paper", "qty": 100, "size": { "h": 8.5, "w": 11, "uom": "in" }, "status": "D" },

{ "item": "planner", "qty": 75, "size": { "h": 22.85, "w": 30, "uom": "cm" }, "status": "D" },

{ "item": "postcard", "qty": 45, "size": { "h": 10, "w": 15.25, "uom": "cm" }, "status": "A" }

]

1. Now when you are hover to the data we are getting some boxes for updating/editing, Delete, Copy, Clone Document.
2. Now click on filter field, in this field you can write a query in it and getting filtered data.  
   ex.  
   { status: "A", qty: {$lt: 30} }  
   { $or: [ { status: "A" }, { qty: { $lt: 30 } } ] }  
   { size: { h: 14, w: 21, uom: "cm" } }
3. Queries are as same as SQL queries.
4. Now click on FIND.  
     
   

14. There is no Primary Key and Foreign key concept in MongoDB  
When we create an object the object ID is taken as a primary Key in MongoDB.  
And There is no Foreign Key in MongoDB, but if we want to implement that part, we need to do in Backed side.  
Ex.  


[MongoDB Shell]

For MongoDB Shell we need to Add path in of Mongosh in Environment variable.  
The mongosh path is in Installed File is  
%APPDATA%/mongodb/mongosh

1. After the installation is done Now open the MongoDB shell.
2. Connect the shell using Connection String, which is provided by MongoDB compass or User’s String.
3. Now you need to use database for performing queries.  
   a. use databasename -: for using Database
4. CRUD queries in MongoDB shell  
    **1.Insert**  
    db.collecion.insertOne({})  
   db.collection.insertMany({})  
   **ex.**  
   db.inventory.insertMany( [

{ item: "journal", qty: 25, size: { h: 14, w: 21, uom: "cm" }, status: "A" },

{ item: "notebook", qty: 50, size: { h: 8.5, w: 11, uom: "in" }, status: "P" },

{ item: "paper", qty: 100, size: { h: 8.5, w: 11, uom: "in" }, status: "D" },

{ item: "planner", qty: 75, size: { h: 22.85, w: 30, uom: "cm" }, status: "D" },

{ item: "postcard", qty: 45, size: { h: 10, w: 15.25, uom: "cm" }, status: "A" },

] );  
**2.Query**  
db.collection.find({})  
**3.Update**  
 db.collection.updateOne({<filter>,<update>,<options>})  
**ex.**   
db.demo3.updateOne( { StudentName: "Parth" }, { $set: { "Address":{$PermanentAddress: "Ahmedabad" }} })  
  
 db.collection.updateMany({<filter>,<update>,<options >})  
 db.collection.replaceone({<filter>,<update>,<options>})  
**4.Delete**  
 db.collection.deleteOne()  
**ex.**db.inventory.deleteOne( { status: "D" } )  
  
 db.collection.deleteMany()  
**ex.**  
db.inventory.deleteMany({ status : "A" })  
   
 db.collection.remove()  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
**[Additional Methods]**

**5.** db.collection.bulkWrite()  
All the DML operations are Performed using bulkWrite.  
IN SQL we are using SP to perform the bulk of operation in MongoDB we are Using Bulk Write For the Same.  
**Ex.**try {

db.pizzas.bulkWrite( [

{ insertOne: { document: { \_id: 3, type: "beef", size: "medium", price: 6 } } },

{ insertOne: { document: { \_id: 4, type: "sausage", size: "large", price: 10 } } },

{ updateOne: {

filter: { type: "cheese" },

update: { $set: { price: 8 } }

} },

{ deleteOne: { filter: { type: "pepperoni"} } },

{ replaceOne: {

filter: { type: "vegan" },

replacement: { type: "tofu", size: "small", price: 4 }

} }

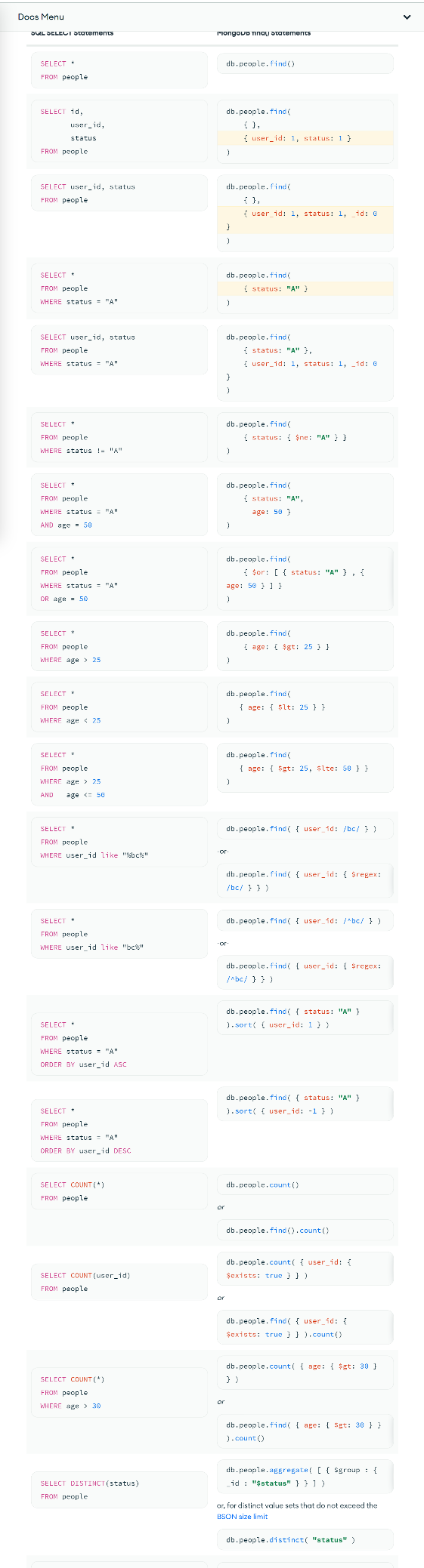
] )

} catch( error ) {

print( error )

}  
db.collection.findOneAndDelete()  
db.collection.findAndModify()

**SQL and MongoDB Comparison  
**

****

[Reference Link]

MongoDB Official Documentation-:

<https://www.mongodb.com/docs/manual/tutorial/query-embedded-documents/>

Concept of Primary Key in MongoDB: <https://www.guru99.com/mongodb-objectid.html>

Crud Operation in MongoDB-: <https://www.mongodb.com/docs/manual/tutorial/query-embedded-documents/>

Simple Crud in MongoDB-: <https://www.mongodb.com/docs/mongodb-shell/crud/read/>

Creating Aggregation Pipeline (used to make Store Procedure):

<https://www.mongodb.com/docs/compass/current/aggregation-pipeline-builder/>,

<https://www.mongodb.com/features/stored-procedures#:~:text=Does%20MongoDB%20support%20stored%20procedures,provides%20the%20aggregation%20pipelines%20framework>.

Creating View in MongoDB:  
<https://www.mongodb.com/docs/compass/current/views/#std-label-view-details>

Connect MongoDB to Provided Connection String-:  
<https://www.mongodb.com/docs/compass/current/connect/#std-label-connect-run-compass>

MongoDB and SQL Comparison:  
<https://www.mongodb.com/docs/manual/reference/sql-comparison/>

Bulk Write Operations-:  
<https://www.mongodb.com/docs/manual/core/bulk-write-operations/>