MongoDB Cheat Sheet

Basic commands

1. Connect to MongoDB: Different ways to connect using Mongoshell

```
mongosh "URI"
mongosh --host mongodb0.example.com --port 28015
mongosh "mongodb://mongodb0.example.com:28015" --username alice --authenticationDatabase admin
```

1. Show databases
   `show dbs`
2. Switch database
   `use <database_name>`
3. Create a collection
   `db.createCollection("<collection_name>")`
4. Show collections in the current database
   `show collections`
5. Insert a document
   `db.<collection_name>.insert({ field1: value1, field2: value2, ... })`
6. Insert multiple documents
   `db.<collection_name>.insertMany([document1, document2, ...])`
7. Find documents
   `db.<collection_name>.find()`

Querying

1. Filter documents with a query
   `db.<collection_name>.find({ field: value })`
2. Equality query
   `db.<collection_name>.find({ field: "value" })`
3. Range query

```
db.<collection_name>.find({ field: { $lt: value } })
db.<collection_name>.find({ field: { $gt: value } })
db.<collection_name>.find({ field: { $lt: value, $gt: value } })
```

4. AND query
   `db.<collection_name>.find({ field1: value1, field2: value2 })`
5. OR query
   `db.<collection_name>.find({ $or: [ { field1: value1 }, { field2: value2 } ] })`
6. Sort ascending
   `db.<collection_name>.find().sort({ field: 1 })`
7. Sort descending
   `db.<collection_name>.find().sort({ field: -1 })`

Update and delete

1. Update documents

```
db.<collection_name>.updateOne({ field: value }, { $set: { new_field: new_value } })
db.<collection_name>.updateMany({ field: value }, { $set: { new_field: new_value } })
```

2. Delete documents

```
db.<collection_name>.deleteOne({ field: value })
db.<collection_name>.deleteMany({ field: value })
```

Aggregation

1. Aggregation pipeline

```
db.<collection_name>.aggregate([
{ $match: { field: value } },
{ $group: { _id: "$field", total: { $sum: 1 } } }
])
```

Indexing

1. Create a single field index
   `db.<collection_name>.createIndex({ field: 1 })`
2. Create a compound index
   `db.<collection_name>.createIndex({ field: 1, another_field: 1 })`
3. List all indexes
   `db.<collection_name>.getIndexes()`

Export and import data

1. Export data to JSON
   `mongoexport --db <database_name> --collection <collection_name> --out <output_file.json>`
2. Import data from JSON
   `mongoimport --db <database_name> --collection <collection_name> --file <input_file.json>`

Author(s):

Muhammad Yahya