

# THAO TÁC VỚI MONGODB

---

## CƠ BẢN

Liệt kê tất cả các database: `show dbs`

Chuyển quyền sử dụng sang một database (không có thì tạo mới): `use dbName`

Hiển thị tất cả các collections có trong database đó: `show collections`

Hiển thị trợ giúp: `help`

## CREATE OPERATIONS

Dùng các lệnh sau

`db.collection.insertOne()` để chèn 1 document

`db.collection.insertMany()` để chèn nhiều document

Ví dụ:

```
db.users.insertOne(  ← collection
{
  name: "sue",        ← field: value
  age: 26,            ← field: value
  status: "pending"   ← field: value } document
})
```

## READ OPERATIONS

Sử dụng lệnh

`db.collection.find()`

để lấy hết tất cả các document. Tuy nhiên ta có thể giới hạn kết quả theo các tiêu chí

Ví dụ

```
db.users.find(  ← collection
{ age: { $gt: 18 } },  ← query criteria
{ name: 1, address: 1 }  ← projection
).limit(5)  ← cursor modifier
```

## UPDATE OPERATIONS

Sử dụng các lệnh sau

`db.collection.updateOne()`

`db.collection.updateMany()`

`db.collection.replaceOne()`

Ví dụ

```
db.users.updateMany(  ← collection
{ age: { $lt: 18 } },  ← update filter
{ $set: { status: "reject" } }  ← update action
)
```

Delete Operations


Sử dụng các lệnh sau

db.collection.deleteOne()

db.collection.deleteMany()

Ví dụ

```
db.users.deleteMany(  
  { status: "reject" }  
)
```



## IMPORT DATA

Import dữ liệu file json (zips.json) vào mongo

```
mongoimport --host localhost --port 27017 --db sampled --  
collection zips --file zips.json
```

## SQL SERVER VS. MONGODB

Một số khái niệm

SQL Terms/Concepts	MongoDB Terms/Concepts
database	database
table	collection
row	document or BSON document
column	field
index	index
table joins	\$lookup, embedded documents
primary key Specify any unique column or column combination as primary key.	primary key In MongoDB, the primary key is automatically set to the _id field.
aggregation (e.g. group by)	aggregation pipeline See the SQL to Aggregation Mapping Chart.

# BÀI TẬP

---

## BÀI 1

Cho file zips.json. Import vào database có tên zipssdb, collection có tên zips.

Sau đó thực hiện các yêu cầu:

- Hiển thị tất cả các documents
- Chèn thêm 1 document mới
- Tìm các documents có city là PALMER
- Tìm các documents có dân số >100000
- Tìm dân số của thành phố FISHERS ISLAND
- Tìm các thành phố có dân số từ 10 - 50
- Tìm tất cả các thành phố của bang MA có dân số trên 500
- Tìm tất cả các bang (distinct)
- Tìm tất cả các bang mà có chứa ít nhất 1 thành phố có dân số trên 100000
- Tính dân số trung bình của mỗi bang
- Bang WA có bao nhiêu city
- Tính số city của mỗi bang
- Tìm tất cả các bang có tổng dân số trên 10000000

## BÀI 2:

Lập trình với mongodb sync driver

```
<dependencies>
  <dependency>
    <groupId>org.mongodb</groupId>
    <artifactId>mongodb-driver-sync</artifactId>
    <version>4.1.0</version>
  </dependency>
  <dependency>
    <groupId>org.mongodb</groupId>
    <artifactId>mongodb-driver-legacy</artifactId>
    <version>4.1.0</version>
  </dependency>
  <!-- https://mvnrepository.com/artifact/com.google.code.gson/gson -->
  <dependency>
    <groupId>com.google.code.gson</groupId>
    <artifactId>gson</artifactId>
    <version>2.8.5</version>
  </dependency>
</dependencies>
```

## BÀI 3:

Lập trình với MongoDB Reactive Streams Driver

```
<dependencies>
  <dependency>
    <groupId>org.mongodb</groupId>
```

```
        <artifactId>mongodb-driver-reactivestreams</artifactId>
        <version>4.1.0</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/com.google.code.gson/gson -->
    <dependency>
        <groupId>com.google.code.gson</groupId>
        <artifactId>gson</artifactId>
        <version>2.8.5</version>
    </dependency>
</dependencies>
```