**MONGODB 03 – THAO TÁC DỮ LIỆU NÂNG CAO VỚI MONGODB**

I – HIỂN THỊ DỮ LIỆU NÂNG CAO TRONG MONGODB

**1. Sử dụng toán tử so sánh trong MongoDB**

|  |  |  |  |
| --- | --- | --- | --- |
| **Phép Toán** | **Cú Pháp** | **Ví dụ** | **Câu lệnh tương ứng trong SQL** |
| Bằng (Equality) | {key: value} | db.admin.find({name: "Tuan Hoang" }).pretty() | .. WHERE name = " Vietpro" |
| Nhỏ hơn (Less Than ) | {key: {$lt: value}} | db.admin.find({age: { $lt: 18}).pretty() | ... WHERE age < 18 |
| Nhỏ hơn bằng (Less Than Equals) | {key: {$lte: value}} | db.admin.find({age: { $lte: 18}).pretty() | ... WHERE age <= 18 |
| Lơn hơn (Greater Than) | {key: {$gt: value}} | db.admin.find({age: { $gt: 12}).pretty() | ... WHERE age > 12 |
| Lớn hơn bằng (Greater Than Equals) | {key: {$gte: value}} | db.admin.find({age: { $gte: 12}).pretty() | ... WHERE age >= 12 |
| Khác (Not Equals) | {key: {$ne: value}} | db.admin.find({age: { $ne: 12}).pretty() | ... WHERE age != 12 |
| Trong ( In) | {key: {$in: [value1, value2,..]}} | db.admin.find({age: { $in: [12, 18]}).pretty() | ... WHERE age IN (12, 18) |
| Không Thuộc (Not In) | {key: {$nin: [value1, value2,..]}} | db.admin.find({age: { $nin: [12, 18]}).pretty() | ... WHERE age NOT IN (12, 18) |

pretty() : trả về kết quả kiểu format dễ nhìn hơn.

**2. AND (,)**

Kết quả của việc lọc phải thỏa mãn tất cả các tiêu chí (giao của nhiều tập hợp trong toán học)

**Cú pháp:**

|  |
| --- |
| db.Category.find({cat\_name:"OPPO", \_id:ObjectId("5dd10d55c674bcd56bb97b23")}) |

**Trong đó:**

* **WHERE** được sử dụng theo dạng đối tượng mà các biểu thức giao nhau trong đó sẽ phân cách với nhau bới dấu “,”

**3. OR**

Kết quả của việc lọc chỉ cần thỏa mãn một trong các tiêu chí (hợp của nhiều tập hợp trong toán học)

**Cú pháp:**

|  |
| --- |
| db.Category.find({    $or: [  {cat\_name: "iPhone"},  {cat\_name: "Samsung"},  {cat\_name: "OPPO"}  ]  }) |

**Trong đó:**

* **$or** đóng vai trò như một Key scope để lưu trữ một mảng các điều kiện

**4. SORT**

**Cú pháp:**

|  |
| --- |
| // Sắp xếp tăng dần  db.Category.find().sort({cat\_name: 1})  // Sắp xếp giảm dần  db.Category.find().sort({cat\_name: -1}) |

**Trong đó:**

* **cat\_name: 1** là sắp xếp theo chiều tăng dần giá trị của thuộc tính cat\_name
* **cat\_name: -1** là sắp xếp theo chiều giảm dần giá trị của thuộc tính cat\_name

**5. LIMIT (lấy ra 1 nhóm data trong danh sách các document)**

**Cú pháp:**

|  |
| --- |
| db.Category.find().limit(3) |

**Trong đó:**

* 3 là số lượng Document được lấy ra từ một Collection (luôn luôn lấy từ đầu)

***Chú ý:***

* *Nếu không muốn lấy ra các Document từ vị trí đầu, hay nói cách khác là bỏ qua một số Document nào đó để lấy bắt đầu ở một vị trí bất kỳ (Giống như LIMIT trong MySQL) thì chúng ta có thể kết hợp với phương thức skip()*

**Cú pháp:**

|  |
| --- |
| db.Category.find().skip(1).limit(3) |

skip: lấy từ đâu

limit: lấy bao nhiêu..

vd: lấy từ vị trí 1 và lấy 3 phần tử.

**Trong đó:**

* 1 là số lượng Document bị bỏ qua

**6. JOIN**

**Cú pháp:**

|  |
| --- |
| db.Product.aggregate([  {  $lookup: {  from: "Category", // tới collection nào  localField: "cat\_id", // trường mới của Product lưu Id của Category  foreignField: "\_id", // lấy id của collection Category  as: "CategoryList" // kết quả được lưu vào một bảng tạm  }  }  ]) |

**Trong đó:**

* **Product** (Collection 1) sẽ là Collection mà mình sẽ hiển thị hết dữ liệu trước
* **Category** (Collection 2) sẽ là Collection hiển thị dữ liệu tương ứng theo Product
* **cat\_id** là thông tin chung của 2 Collection nằm ở Collection 1
* **\_id** là thông tin chung của 2 Collection nằm ở Collection 2

II – IMPORT VÀ EXPORT CSDL TRONG MONGODB

Không giống như CSDL quan hệ cho phép chúng ta Import và Export cả Database, CSDL không quan hệ chỉ cho phép Import và Export từng Collection một.

**1. Import**

**Cú pháp:**

|  |
| --- |
| mongoimport --db vietpro\_mongodb --collection users --file users.json |

**Trong đó:**

* **vietpro\_mongodb**: CSDL chúng ta muốn Import
* **User** sau **–collection** chính là Collection mà chúng ta muốn import (chọn collection mà chúng ta muốn import)
* **User.json** sau **–file** chính là tên file .JSON chứa Collection mà chúng ta muốn import(thêm vào) vào

**2. Export**

**Cú pháp:**

|  |
| --- |
| mongoexport --db vietpro\_mongodb --collection users --out users.json |

**Trong đó:**

* **vietpro\_mongodb**: CSDL chúng ta muốn Export
* **User** sau **–collection** chính là Collection mà chúng ta muốn export
* **User** sau **–out** chính là tên file .JSON chứa Collection mà chúng ta vừa export(xuất ra) thành công

**3. Dump (xuất tất cả database)**

**Cú pháp:**

|  |
| --- |
| mongodump --db vietpro\_mongodb |

**Trong đó:**

* **vietpro\_mongodb**: Export nguyên CSDL là vietpro\_mongodb (là tên của CSDL)

**4. Restore (chạy ngang hàng với dump)**

**Cú pháp:**

|  |
| --- |
| mongorestore --db vietpro\_mongodb dump/vietpro\_mongodb |

**Trong đó:**

* **vietpro\_mongodb**: Đặt tên CSDL mà chúng ta muốn Restore
* **dump/vietpro\_mongodb**: đường dẫn đến các file dữ liệu đã dump trước đó

**NOTE: muốn sản phẩm nằm trong danh mục nào 🡺 lấy id của danh mục đó đưa vào cho sản phẩm tương ứng để làm khóa ngoại**

* **lấy id của danh mục samsung đưa vào cho sản phẩm samsung làm khóa ngoại**