

# Try It Yourself \_Aggregation Practice

17 Tháng Mười 2025 7:49 CH

## Try It Yourself – MongoDB Aggregation Practice

### Step 1 – Create Database and Collection

Command (Lệnh):

```
js

use Inventory
db.createCollection("sales_invent")
```

Explanation (Giải thích):

This command creates a new database named **Inventory** and a collection named **sales\_invent** to store sales records.

(Lệnh này tạo một database mới tên **Inventory** và collection **sales\_invent** để lưu dữ liệu bán hàng.)

### Step 2 – Insert Sample Documents

Command:

```
db.sales_invent.insertMany([
  { customername: "Richard", gender: "M", purchased_product: "Cereals", quantity: 6, price: 60 },
  { customername: "Williams", gender: "M", purchased_product: "Vegetables", quantity: 10, price: 150 },
  { customername: "Emma", gender: "F", purchased_product: "Fruits", quantity: 8, price: 200 },
  { customername: "John", gender: "M", purchased_product: "Baby Food", quantity: 3, price: 300 },
  { customername: "Smith", gender: "M", purchased_product: "Fruits", quantity: 5, price: 180 }
])
```

Explanation:

We insert five sales documents with customer details, gender, product, quantity, and price.

(Chèn 5 bản ghi khách hàng chứa thông tin: tên, giới tính, sản phẩm, số lượng và giá.)

👉 Đây là dữ liệu nền cho các bài tập aggregation.


### Step 3 – Display Only Selected Fields and Limit Documents

(Bước 3 – Hiển thị một số trường và giới hạn số document)

Command:

```
js

db.sales_invent.aggregate([
  { $project: { _id: 0, customername: 1, purchased_product: 1, price: 1 } },
  { $limit: 10 }
])
```

 Sao chép mã

#### Explanation:

- \$project chọn chỉ 3 trường (customername, purchased\_product, price) và loại bỏ \_id.
- \$limit: 10 chỉ lấy tối đa 10 dòng đầu tiên.
  - Giống như lệnh **SELECT TOP 10 customername, product, price** trong SQL.

#### Step 4 – Group by Product and Calculate Total Sale

(Bước 4 – Nhóm theo sản phẩm và tính tổng doanh thu)

#### Command:

```
js                                                                    Sao chép mã

db.sales_invent.aggregate([
  { $group: {
    _id: "$purchased_product",
    total_sale: { $sum: { $multiply: ["$price", "$quantity"] } }
  }},
  { $match: { total_sale: { $gte: 500 } } }
])
```

#### Explanation:

- \$group nhóm dữ liệu theo purchased\_product.
- \$sum tính tổng doanh thu, công thức: price × quantity.
- \$match lọc chỉ các sản phẩm có tổng doanh thu ≥ 500.
  - Giúp bạn tìm **mặt hàng bán chạy nhất**.

#### Step 5 – Sort by Customer Name and Limit Results

(Bước 5 – Sắp xếp tên khách hàng và giới hạn kết quả)

#### Command:

```
js                                                                    Sao chép mã

db.sales_invent.aggregate([
  { $project: { customername: 1, purchased_product: 1, _id: 0 } },
  { $sort: { customername: 1 } },
  { $limit: 3 }
])
```

#### Explanation:

- \$sort: { customername: 1 } → sắp xếp theo tên tăng dần (A–Z).
- \$limit: 3 → chỉ hiển thị 3 dòng đầu tiên.
  - Thích hợp để **hiển thị danh sách khách hàng đầu tiên** theo tên bảng chữ cái.

#### Step 6 – Add a New Field to All Documents

(Bước 6 – Thêm trường mới vào tất cả document)

#### Command:

```
js
```

```
db.sales_invent.aggregate([
  { $addField: { product_type: "edibles" } }
])
```

**Explanation:**

- \$addFields thêm trường product\_type với giá trị "edibles" cho mọi document.  
→ Giống như thêm cột "loại hàng hóa" cho tất cả sản phẩm.

Nếu bạn muốn lưu lại thay đổi này, dùng thêm \$out:

```
js
{ $out: "sales_invent_updated" }
```

### Step 7 – Group and Write Output to Another Collection

(Bước 7 – Nhóm dữ liệu và ghi ra collection mới)

**Command:**

```
js
db.sales_invent.aggregate([
  { $group: { _id: "$purchased_product", Total_quantity: { $sum: "$quantity" } } },
  { $out: "Product_report" }
])
```

**Explanation:**

- \$group cộng dồn số lượng (Total\_quantity) theo từng sản phẩm.
- \$out ghi kết quả ra collection **Product\_report** trong cùng database.  
→ Kết quả giống như xuất báo cáo "Tổng số lượng theo sản phẩm".

### Step 8 – Calculate Total Price (price × quantity)

(Bước 8 – Tính tổng tiền từng giao dịch)

**Command:**

```
js
db.sales_invent.aggregate([
  { $addField: { total_price: { $multiply: ["$price", "$quantity"] } } },
  { $sort: { _id: -1 } },
  { $limit: 3 }
])
```

**Explanation:**

- \$multiply nhân giá và số lượng để ra total\_price.

- \$limit: 3 hiển thị 3 bản ghi cuối cùng (có thể thay bằng \$skip để chọn đoạn khác).  
→ Thường dùng để tính doanh thu từng hóa đơn.

### Step 9 – Display Products with Quantity > 5

(Bước 9 – Lọc sản phẩm có số lượng mua > 5)

Command:

```
js
db.sales_invent.aggregate([
  { $match: { quantity: { $gt: 5 } } },
  { $project: { customername: 1, purchased_product: 1, quantity: 1, _id: 0 } }
])
```

Explanation:

- \$match lọc document có quantity > 5.
- \$project chọn trường cần hiển thị.  
→ Tương đương với **SELECT ... WHERE quantity > 5** trong SQL.

### Step 10 – Concatenate Customer and Product Name

(Bước 10 – Ghép tên khách hàng và tên sản phẩm)

Command:

```
js
db.sales_invent.aggregate([
  { $match: { gender: "M" } },
  { $addFields: {
    customer_detail: { $concat: ["$customername", " - ", "$purchased_product"] }
  }},
  { $project: { customer_detail: 1, gender: 1, _id: 0 } }
])
```

Explanation:

- \$match chọn khách hàng nam (M).
- \$concat nối hai chuỗi bằng dấu “-”.  
→ Kết quả: "Richard - Cereals", "Williams - Vegetables", "John - Baby Food", v.v.  
→ Giúp tạo **tên hiển thị tùy chỉnh** cho báo cáo hoặc UI.

### ✓ Summary for Practice

Stage	Purpose	Vietnamese Meaning
\$project	Choose fields	Chọn trường hiển thị
\$match	Filter docs	Lọc dữ liệu
\$group	Aggregate data	Nhóm và tính toán
\$addFields	Add computed fields	Thêm cột mới
\$sort	Sort result	Sắp xếp kết quả
\$limit / \$skip	Limit or skip output	Giới hạn / bỏ qua dòng

\$out	Write to collection	Ghi ra collection mới
-------	---------------------	-----------------------