

# MongoDB Aggregation Pipeline Guide

## دليل مراحل ومشغلات خط أنابيب التجميع في MongoDB

xAI

آي أي إكس

May 22, 2025

22 مايو 2025

## Introduction

### مقدمة

This document provides a comprehensive guide to the MongoDB aggregation pipeline, focusing on its main stages and operators. It includes explanations and examples to help users understand how to process and transform data efficiently.

الرئيسية المراحل على التركيز مع MongoDB، في التجميع أنابيب لخط شاملاً دليلاً المستند هذا يوفر وتحويلها البيانات معالجة كيفية فهم على المستخدمين لمساعدة وأمثلة شروحات يتضمن والمشغلات. بكفاءة.

## Aggregation Pipeline Stages

### التجميع أنابيب خط مراحل

The MongoDB aggregation pipeline consists of stages that process documents sequentially. Each stage transforms the input documents and passes the results to the next stage.

كل متسلسل. بشكل المستندات تعالج مراحل من MongoDB في التجميع أنابيب خط يتكون التالية. المرحلة إلى النتائج وتكرر المدخلة المستندات تحول مرحلة

Stage	Description
المرحلة	الوصف
\$match	Filters documents based on specified conditions, similar to WHERE in SQL. يصفى المستندات بناءً على شروط محددة، مشابه لـ WHERE في SQL.
\$group	Groups documents by a key and applies aggregate functions, like GROUP BY in SQL. يجمع المستندات حسب مفتاح ويطبق دوال إحصائية، مشابه لـ BY, GROUP
\$project	Selects specific fields or computed values to include in the output, like SELECT in SQL. ينتقي حقولاً محددة أو قيم محسوبة لتضمينها في الإخراج، مشابه لـ SELECT.
\$sort	Sorts documents based on a specified field. يرتب المستندات بناءً على حقل محدد.
\$limit	Restricts the number of documents returned. يحد من عدد المستندات المرجعة.
\$skip	Skips a specified number of documents, useful for pagination. يتجاوز عدداً محددًا من المستندات، مفيد للتصفح.
\$unwind	Deconstructs an array field to output a document for each element. يفكك حقل المصفوفة لإخراج مستند لكل عنصر.
\$lookup	Performs a left outer join with another collection. يؤدي عملية ربط خارجي أيسر مع مجموعة أخرى.
\$addField	Adds new fields to documents, often computed. يضيف حقولاً جديدة إلى المستندات، غالباً محسوبة.
\$count	Counts the number of documents in the pipeline.

## Example: Using *matchandgroup*

### *matchgroup* استخدام مثال:

Suppose we have a collection `orders` with documents like:

```
{ customer: "Ali", amount: 100, date: "2025-01-01" }
{ customer: "Sara", amount: 200, date: "2025-01-02" }
{ customer: "Ali", amount: 150, date: "2025-01-03" }
```

To find the total amount spent by each customer in January 2025:

```
db.orders.aggregate([
  { $match: { date: { $regex: "^2025-01" } } },
  { $group: { _id: "$customer", total: { $sum: "$amount" } } }
])
```

Output:

```
{ "_id": "Ali", "total": 250 }
{ "_id": "Sara", "total": 200 }
```

مثال: للعثور على المبلغ الإجمالي الذي أنفقه كل عميل في يناير 2025.

## Group Operators

### التجميع مشغلات

Operator	Description
المشغل	الوصف
\$sum	Sums values or counts documents. يجمع القيم أو يحسب عدد المستندات.
\$avg	Computes the average of values. يحسب متوسط القيم.
\$min	Returns the smallest value. يعيد أصغر قيمة.
\$max	Returns the largest value. يعيد أكبر قيمة.
\$push	Collects values into an array. يجمع القيم في مصفوفة.
\$addToSet	Collects unique values into an array. يجمع القيم الفريدة في مصفوفة.
\$first	Returns the first value (based on input order). يعيد القيمة الأولى (حسب ترتيب الإدخال).
\$last	Returns the last value. يعيد القيمة الأخيرة.

## Project and AddFields Operators

### مشغلات *projectaddFields*

Operator	Description	الوصف
المشغل		
\$add	Adds numbers.	يجمع الأرقام.
\$subtract	Subtracts numbers.	يطرح الأرقام.
\$multiply	Multiplies numbers.	يضرب الأرقام.
\$divide	Divides numbers.	يقسم الأرقام.
\$concat	Concatenates strings.	يدمج النصوص.
\$toUpper	Converts text to uppercase.	يحول النص إلى حروف كبيرة.
\$toLower	Converts text to lowercase.	يحول النص إلى حروف صغيرة.
\$substr	Extracts a substring.	يقطع النصوص.
\$arrayElemAt	Returns an element from an array at a specific index.	يجيب عنصراً من مصفوفة في فهرس معين.
\$ifNull	Replaces null with a default value.	يعوض null بقيمة افتراضية.
\$cond	Conditional logic (if/else)	منطق شرطي (fi/else).

## Example: Using *project*

### : *project*

To format a customer's name in uppercase and calculate a discounted amount:

```
db.orders.aggregate([
  { $project: {
    customer: { $toUpper: "$customer" },
    discountedAmount: { $multiply: ["$amount", 0.9] }
  } }
])
```

Output:

```
{ "customer": "ALI", "discountedAmount": 90 }
{ "customer": "SARA", "discountedAmount": 180 }
{ "customer": "ALI", "discountedAmount": 135 }
```

مثال: لتنسيق اسم العميل بحروف كبيرة وحساب مبلغ مخفض.

## Conclusion

### خاتمة

The MongoDB aggregation pipeline is a powerful tool for data processing and analysis. By combining stages and operators, users can filter, group, transform, and analyze data efficiently. المراحل بين الجمع خلال من وتحليلها. البيانات لمعالجة قوية أداة MongoDB في التجميع أنابيب خط بكفاءة. وتحليلها تحويلها، تجميعها، البيانات، تصفية للمستخدمين يمكن والمشغلات،