

MongoDB : sorting documents by multiple fields

To sort documents by multiple fields in the MongoDB shell, use the `sort()` method after your `find()` query. The `sort()` method accepts a document where keys are the field names and values are either `1` for ascending order or `-1` for descending order. The order of fields within the `sort()` document determines the sorting priority.

Here's an example:

Assume a collection named `students` with documents like:

Code

```
{ "name": "Alice", "grade": "A", "age": 18 }
{ "name": "Bob", "grade": "B", "age": 19 }
{ "name": "Charlie", "grade": "A", "age": 17 }
{ "name": "David", "grade": "B", "age": 18 }
```

To sort these documents first by `grade` in ascending order, and then by `age` in descending order for students with the same grade:

JavaScript

```
db.students.find().sort({ grade: 1, age: -1 });
```

Explanation:

`db.students.find()`: This retrieves all documents from the `students` collection.

`.sort({ grade: 1, age: -1 })`: This applies the sorting criteria.

`grade: 1`: Documents are primarily sorted by the `grade` field in ascending order (e.g., "A" before "B").

`age: -1`: For documents that have the same `grade`, they are then sorted by the `age` field in descending order (e.g., 19 before 18).

Example Output (based on the sample data):

Code

```
{ "name": "Charlie", "grade": "A", "age": 17 }  
{ "name": "Alice", "grade": "A", "age": 18 }  
{ "name": "Bob", "grade": "B", "age": 19 }  
{ "name": "David", "grade": "B", "age": 18 }
```

Revision #2

Created 29 October 2025 02:43:35 by AI API

Updated 1 December 2025 14:01:08 by AI Channel