

MongoDB CRUD Operations > Delete Documents

Delete Documents

This page provides examples in:

MONGO SHELL COMPASS PYTHON JAVA (SYNC) NODEJS PHP MOTOR
JAVA (ASYNC) C# PERL RUBY SCALA GO

This page uses the following `mongo` shell methods:

- `db.collection.deleteMany()`
- `db.collection.deleteOne()`

The examples on this page use the `inventory` collection. To populate the `inventory` collection, run the following:

```
db.inventory.insertMany( [  
  { item: "journal", qty: 25, size: { h: 14, w: 21, uom: "cm" }, status: "A" },  
  { item: "notebook", qty: 50, size: { h: 8.5, w: 11, uom: "in" }, status: "P" },  
  { item: "paper", qty: 100, size: { h: 8.5, w: 11, uom: "in" }, status: "D" },  
  { item: "planner", qty: 75, size: { h: 22.85, w: 30, uom: "cm" }, status: "D" },  
  { item: "postcard", qty: 45, size: { h: 10, w: 15.25, uom: "cm" }, status: "A" },  
] );
```

You can run the operation in the web shell below:

Full

Reset

Clear

Delete All Documents

To delete all documents from a collection, pass an empty filter document `{}` to the `db.collection.deleteMany()` method.

The following example deletes *all* documents from the `inventory` collection:

```
db.inventory.deleteMany({})
```

The method returns a document with the status of the operation. For more information and examples, see `deleteMany()`.

Delete All Documents that Match a Condition

You can specify criteria, or filters, that identify the documents to delete. The filters use the same syntax as read operations.

To specify equality conditions, use `<field>:<value>` expressions in the query filter document:

```
{ <field1>: <value1>, ... }
```

A query filter document can use the query operators 1

[Search Documentation](#)

```
{ <field1>: { <operator1>: <value1> }, ... }
```

To delete all documents that match a deletion criteria, pass a filter parameter to the `deleteMany()` method.

The following example removes all documents from the `inventory` collection where the `status` field equals `"A"`:

```
db.inventory.deleteMany({ status : "A" })
```

The method returns a document with the status of the operation. For more information and examples, see `deleteMany()`.

Delete Only One Document that Matches a Condition

To delete at most a single document that matches a specified filter (even though multiple documents may match the specified filter) use the `db.collection.deleteOne()` method.

The following example deletes the *first* document where `status` is `"D"`:

```
db.inventory.deleteOne( { status: "D" } )
```

Delete Behavior

Indexes

Delete operations do not drop indexes, even if deleting all documents from a collection.

Atomicity

All write operations in MongoDB are atomic on the level of a single document. For more information on MongoDB and atomicity, see [Atomicity and Transactions](#).

Write Acknowledgement

 mongoDB. Documentation ▼

Search Documentation

With write concerns, you can specify the level of acknowledgement requested from MongoDB for write operations. For details, see [Write Concern](#).

SEE ALSO:

- `db.collection.deleteMany()`
- `db.collection.deleteOne()`
- [Additional Methods](#)