

MongoDB CRUD Operations > Query Documents > Project Fields to Return from Query

Project Fields to Return from Query

This page provides examples in:

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

By default, queries in MongoDB return all fields in matching documents. To limit the amount of data that MongoDB sends to applications, you can include a projection document to specify or restrict fields to return.

This page provides examples of query operations with projection using the `db.collection.find()` method in the mongo shell. The examples on this page use the `inventory` collection. To populate the `inventory` collection, run the following:

```
db.inventory.insertMany( [
  { item: "journal", status: "A", size: { h: 14, w: 21, uom: "cm" }, instock: [ { warehou
  { item: "notebook", status: "A", size: { h: 8.5, w: 11, uom: "in" }, instock: [ { ware
  { item: "paper", status: "D", size: { h: 8.5, w: 11, uom: "in" }, instock: [ { warehou
  { item: "planner", status: "D", size: { h: 22.85, w: 30, uom: "cm" }, instock: [ { ware
  { item: "postcard", status: "A", size: { h: 10, w: 15.25, uom: "cm" }, instock: [ { war
]);
```

You can run the operation in the web shell below:

Full

Reset

Clear

Return All Fields in Matching Documents

If you do not specify a projection document, the `db.collection.find()` method returns all fields in the matching documents.

The following example returns all fields from all documents in the `inventory` collection where the `status` equals "A":

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

The operation corresponds to the following SQL statement:

```
SELECT * from inventory WHERE status = "A"
```

Return the Specified Fields and the `_id` Field Only

A projection can explicitly include several fields by setting the `<field>` to `1` in the projection document. The following operation returns all documents that match the query. In the result set, only the `item`, `status` and, by default, the `_id` fields return in the matching documents.

```
db.inventory.find( { status: "A" }, { item: 1, status: 1, _id: 0 } )
```

[Search Documentation](#)

The operation corresponds to the following SQL statement:

```
SELECT _id, item, status from inventory WHERE status = "A"
```

Suppress _id Field

You can remove the `_id` field from the results by setting it to `0` in the projection, as in the following example:

```
db.inventory.find( { status: "A" }, { item: 1, status: 1, _id: 0 } )
```

The operation corresponds to the following SQL statement:

```
SELECT item, status from inventory WHERE status = "A"
```

NOTE:

With the exception of the `_id` field, you cannot combine inclusion and exclusion statements in projection documents.

Return All But the Excluded Fields

Instead of listing the fields to return in the matching document, you can use a projection to exclude specific fields. The following example which returns all fields except for the `status` and the `instock` fields in the matching documents:

```
db.inventory.find( { status: "A" }, { status: 0, instock: 0 } )
```

NOTE:

With the exception of the `_id` field, you cannot cor documents.

[Search Documentation](#)

Return Specific Fields in Embedded Documents

You can return specific fields in an embedded document. Use the dot notation to refer to the embedded field and set to `1` in the projection document.

The following example returns:

- The `_id` field (returned by default),
- The `item` field,
- The `status` field,
- The `uom` field in the `size` document.

The `uom` field remains embedded in the `size` document.

```
db.inventory.find(  
  { status: "A" },  
  { item: 1, status: 1, "size.uom": 1 }  
)
```

Suppress Specific Fields in Embedded Documents

You can suppress specific fields in an embedded document. Use the dot notation to refer to the embedded field in the projection document and set to `0`.

The following example specifies a projection to exclude the `uom` field inside the `size` document. All other fields are returned in the matching documents:

```
db.inventory.find(
  { status: "A" },
  { "size.uom": 0 }
)
```

[Search Documentation](#)

Projection on Embedded Documents in an Array

Use dot notation to project specific fields inside documents embedded in an array.

The following example specifies a projection to return:

- The `_id` field (returned by default),
- The `item` field,
- The `status` field,
- The `qty` field in the documents embedded in the `instock` array.

```
db.inventory.find( { status: "A" }, { item: 1, status: 1, "instock.qty": 1 } )
```

Project Specific Array Elements in the Returned Array

For fields that contain arrays, MongoDB provides the following projection operators for manipulating arrays: `$elemMatch`, `$slice`, and `$`.

The following example uses the `$slice` projection operator to return the last element in the `instock` array:

```
db.inventory.find( { status: "A" }, { item: 1, status: 1, instock: { $slice: -1 } } )
```

`$elemMatch`, `$slice`, and `$` are the *only* way to project specific elements to include in the returned array. For instance, you *cannot* project specific array elements using the array index; e.g. `{ "instock.0": 1 }` projection will *not* project the array with the first element.

SEE ALSO:

[Query Documents](#)

