



This version of the manual is no longer supported.

MongoDB CRUD Operations > MongoDB CRUD Tutorials > Limit Fields to Return from a Query

Limit Fields to Return from a Query

On this page

- Return All Fields in Matching Documents
- Return the Specified Fields and the `_id` Field Only
- Return Specified Fields Only
- Return All But the Excluded Field
- Return Specific Fields in Embedded Documents
- Projection for Array Fields

The projection document limits the fields to return for all matching documents. The projection document can specify the inclusion of fields or the exclusion of fields.

The specifications have the following forms:

Syntax	Description
<code><field>: <1 or true></code>	Specify the inclusion of a field.
<code><field>: <0 or false></code>	Specify the suppression of the field.

IMPORTANT:

The `_id` field is, by default, included in the result set. To suppress the `_id` field from the result set, specify `_id: 0` in the projection document.

You cannot combine inclusion and exclusion semantics in a single projection with the *exception* of the `_id` field.

This tutorial offers various query examples that limit the fields to return for all matching documents. The examples in this tutorial use a collection `inventory` and use the `db.collection.find()` method in the mongo shell. The `db.collection.find()` method returns a cursor to the retrieved documents. For examples on query selection criteria, see Query Documents.

Return All Fields in Matching Documents

If you specify no projection, the `find()` method returns all fields of all documents that match the query.

```
db.inventory.find( { type: 'food' } )
```

This operation will return all documents in the `inventory` collection where the value of the `type` field is `'food'`. The returned documents contain all its fields.

Return the Specified Fields and the `_id` Field Only

A projection can explicitly include several fields. In the following operation, `find()` method returns all documents that match the query. In the result set, only the `item` and `qty` fields and, by default, the `_id` field return in the matching documents.

```
db.inventory.find( { type: 'food' }, { item: 1, qty: 1 } )
```

Return Specified Fields Only

You can remove the `_id` field from the results by specifying its exclusion in the projection, as in the following example:

```
db.inventory.find( { type: 'food' }, { item: 1, qty: 1, _id:0 } )
```

This operation returns all documents that match the query. In the result set, *only* the `item` and `qty` fields return in the matching documents.

Return All But the Excluded Field

To exclude a single field or group of fields you can use a projection in the following form:



```
db.inventory.find( { type: 'food' }, { type:0 } )
```

This operation returns all documents where the value of the `type` field is `food`. In the result set, the `type` field does not return in the matching documents.

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

Return Specific Fields in Embedded Documents

Use the dot notation to return specific fields inside an embedded document. For example, the `inventory` collection contains the following document:

```
{
  "_id" : 3,
  "type" : "food",
  "item" : "aaa",
  "classification": { dept: "grocery", category: "chocolate" }
}
```

The following operation returns all documents that match the query. The specified projection returns only the `category` field in the `classification` document. The returned `category` field remains inside the `classification` document.

```
db.inventory.find(
  { type: 'food', _id: 3 },
  { "classification.category": 1, _id: 0 }
)
```

The operation returns the following document:



```
{ "classification" : { "category" : "chocolate" } }
```

Projection for Array Fields

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

For example, the `inventory` collection contains the following document:

```
{ "_id" : 5, "type" : "food", "item" : "aaa", "ratings" : [ 5, 8, 9 ] }
```

Then the following operation uses the `$slice` projection operator to return just the first two elements in the `ratings` array.

```
db.inventory.find( { _id: 5 }, { ratings: { $slice: 2 } } )
```

`$elemMatch`, `$slice`, and `$` are the *only* way to project *portions* of an array. For instance, you *cannot* project a portion of an array using the array index; e.g. `{ "ratings.0": 1 }` projection will *not* project the array with the first element.

SEE ALSO:

Query Documents