MongoDB findOne

Summary: in this tutorial, you'll learn how to use the MongoDB `findOne()` method to retrieve a single document from a collection.

Introduction to MongoDB findOne() method

The `findOne()` returns a single document from a collection that satisfies the specified condition.

The `findOne()` method has the following syntax:

```
db.collection.findOne(query, projection)
```

The `findOne()` accepts two optional arguments: `query` and `projection` .

- The `query` is a document that specifies the selection criteria.
- The `projection` is a document that specifies the fields in the matching document that you want to return.

If you omit the `query` , the `findOne()` returns the first document in the collection according to the natural order which is the order of documents on the disk.

If you don't pass the `projection` argument, then `findOne()` will include all fields in the matching documents.

To specify whether a field should be included in the returned document, you use the following format:

```
{field1: value, field1: value, ... }
```

If the `value` is `true` or `1` , MongoDB will include the field in the returned document. In case the `value` is `false` or `0` , MongoDB won't include it.

By default, MongoDB always includes the `_id` field in the returned documents. To suppress it, you need to explicitly specify `_id: 0` in the `projection` argument.

If multiple documents satisfy the `query`, the `findOne()` method returns the first document based on the order of documents stored on the data storage.

> Note that there are other forms of projections such as array projection and aggregation projection which are not covered in this tutorial.

MongoDB findOne() method examples

We'll use the following `products` collection for the demonstration:

```
db.products.insertMany([
    { "_id" : 1, "name" : "xPhone", "price" : 799, "releaseDate": ISODate("2011-0
    { "_id" : 2, "name" : "xTablet", "price" : 899, "releaseDate": ISODate("2011-
    { "_id" : 3, "name" : "SmartTablet", "price" : 899, "releaseDate": ISODate("2
    { "_id" : 4, "name" : "SmartPad", "price" : 699, "releaseDate": ISODate("2020
    { "_id" : 5, "name" : "SmartPhone", "price" : 599,"releaseDate": ISODate("202
])
```

1) Using MongoDB findOne() method with zero argument

The following example uses the `findOne()` method to select the first document from the `products` collection:

```
db.products.findOne()
```

The statement returns all fields of the matching document:

```
{
  _id: 1,
```

```
  name: 'xPhone',
  price: 799,
  releaseDate: ISODate("2011-05-14T00:00:00.000Z"),
  spec: { ram: 4, screen: 6.5, cpu: 2.66 },
  color: [ 'white', 'black' ],
  storage: [ 64, 128, 256 ]
}
```

Note that omitting the `query` is the same as passing the `query` as an empty document:

```
db.products.findOne({})
```

2) Using MongoDB findOne() method with a filter

The following statement uses the `findOne()` method to find the document whose `_id` is 2.

```
db.products.findOne({_id:2})
```

It returns all the fields of the document with `_id` 2:

```
{
  _id: 2,
  name: 'xTablet',
  price: 899,
  releaseDate: ISODate("2011-09-01T00:00:00.000Z"),
  spec: { ram: 16, screen: 9.5, cpu: 3.66 },
  color: [ 'white', 'black', 'purple' ],
  storage: [ 128, 256, 512 ]
}
```

3) Using MongoDB findOne() method to select some fields

The following example uses the `findOne()` method to find the document with `_id 5` . And it returns only the `_id` and `name` fields of the matching document:

```
db.products.findOne({_id: 5}, {name: 1})
```

The query returned the following document:

```
{ "_id" : 5, "name" : "SmartPhone" }
```

As you can see clearly from the output, MongoDB includes the `_id` field in the returned document by default.

To completely remove it from the returned document, you need to explicitly specify `_id:0` in the `projection` document like this:

```
db.products.findOne({ _id: 5 }, {name: 1, _id: 0})
```

It returned the following document:

```
{ "name" : "SmartPhone" }
```

Summary

- Use the `findOne()` method to retrieve the first document in a collection that satisfies the specified condition.