

MongoDB order with Sort() & Limit() Query with Examples

Details

Last Updated: 03 January 2020

What is Query Modifications?

Mongo DB provides query modifiers such as the 'limit' and 'Orders' clause to provide more flexibility when executing queries. We will take a look at the following query modifiers

MongoDB Limit Query Results

This modifier is used to limit the number of documents which are returned in the result set for a query. The following example shows how this can be done.

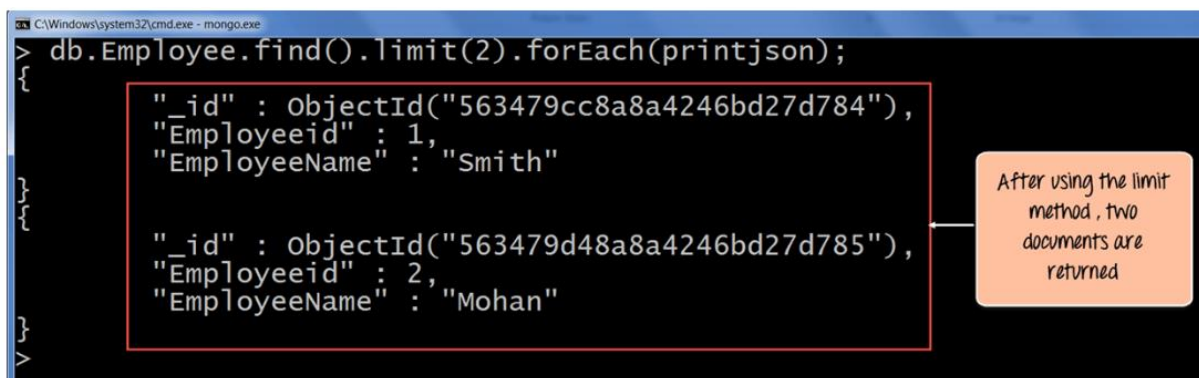
```
db.Employee.find().limit(2).forEach(printjson);
```

Code Explanation:

1. The above code takes the find function which returns all of the documents in the collection but then uses the limit clause to limit the number of documents being returned to just 2.

Output:

If the command is executed successfully, the following Output will be shown



The screenshot shows a terminal window with the command `db.Employee.find().limit(2).forEach(printjson);` executed. The output displays two JSON documents. The first document has an `_id` of `563479cc8a8a4246bd27d784`, an `Employeeid` of `1`, and an `EmployeeName` of `"Smith"`. The second document has an `_id` of `563479d48a8a4246bd27d785`, an `Employeeid` of `2`, and an `EmployeeName` of `"Mohan"`. A red box highlights these two documents. An orange callout box with an arrow pointing to the second document contains the text: "After using the limit method, two documents are returned".

The output clearly shows that since there is a limit modifier, so at most just 2 records are returned as part of the result set based on the ObjectId in ascending order.

MongoDB Sort by Descending Order

One can specify the order of documents to be returned based on ascending or descending order of any key in the collection. The following example shows how this can be done.

```
db.Employee.find().sort({Employeeid:-1}).forEach(printjson)
```

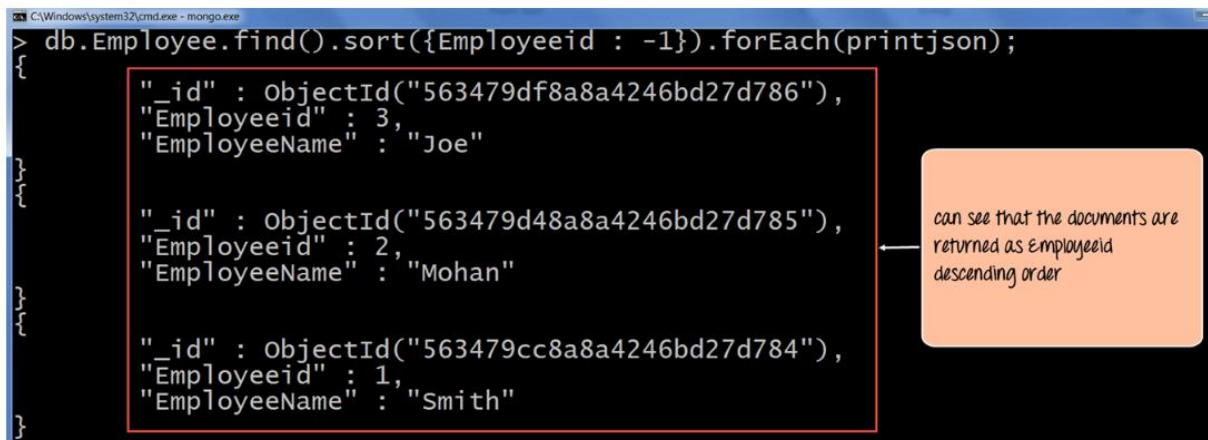
Code Explanation:

1. The above code takes the sort function which returns all of the documents in the collection but then uses the modifier to change the order in which the records are returned. Here the -1 indicates that we want to return the documents based on the descending order of Employee id.

If the command is executed successfully, the following Output will be shown

Output:

Output:



```
C:\Windows\system32\cmd.exe - mongo.exe
> db.Employee.find().sort({Employeeid : -1}).forEach(printjson);
{"_id" : ObjectId("563479df8a8a4246bd27d786"),
 "Employeeid" : 3,
 "EmployeeName" : "Joe"}
{"_id" : ObjectId("563479d48a8a4246bd27d785"),
 "Employeeid" : 2,
 "EmployeeName" : "Mohan"}
{"_id" : ObjectId("563479cc8a8a4246bd27d784"),
 "Employeeid" : 1,
 "EmployeeName" : "Smith"}
```

can see that the documents are returned as Employeeid descending order

The output clearly shows the documents being returned in descending order of the Employeeid. Ascending order is defined by value 1.