MongoDB Indexing - createIndex(), dropindex()

Indexes are very important in any database, and with MongoDB it's no different. With the use of Indexes, performing queries in MongoDB becomes more efficient.

If you had a collection with thousands of documents with no indexes, and then you query to find certain documents, then in such case MongoDB would need to scan the entire collection to find the documents. But if you had indexes, MongoDB would use these indexes to limit the number of documents that had to be searched in the collection.

Indexes are special data sets which store a partial part of the collection's data. Since the data is partial, it becomes easier to read this data. This partial set stores the value of a specific field or a set of fields ordered by the value of the field.

Understanding Impact of Indexes

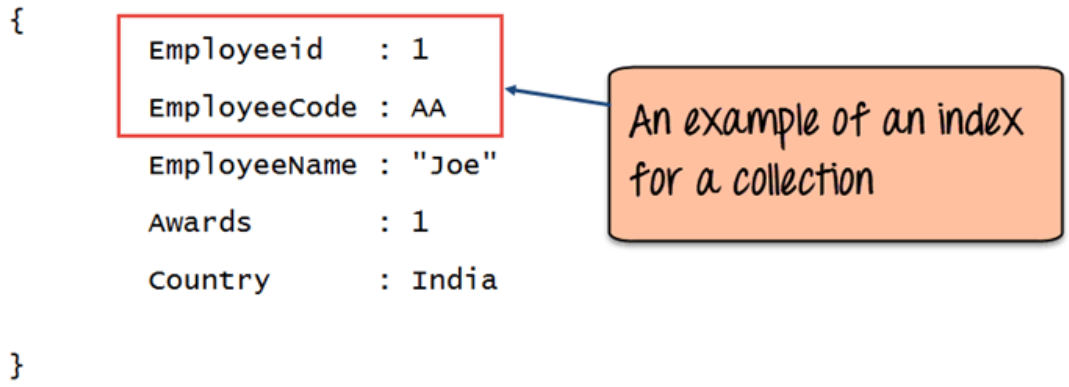
Now even though from the introduction we have seen that indexes are good for queries, but having too many indexes can slow down other operations such as the Insert, Delete and Update operation.

If there are frequent insert, delete and update operations carried out on documents, then the indexes would need to change that often, which would just be an overhead for the collection.

The below example shows an example of what field values could constitute an index in a collection. An index can either be based on just one field in the collection, or it can be based on multiple fields in the collection.

In the example below, the Employeeid "1" and EmployeeCode "AA" are used to index the documents in the collection. So when a query search is made, these indexes will be used to quickly and efficiently find the required documents in the collection.

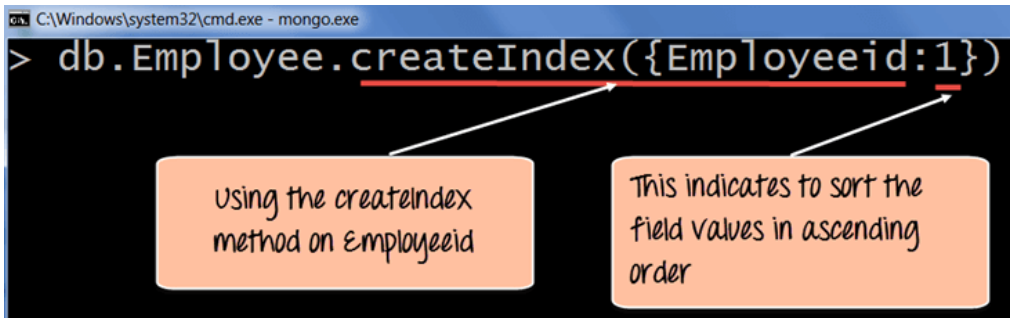
So even if the search query is based on the EmployeeCode "AA", that document would be returned.



How to Create Indexes: createIndex()

Creating an Index in MongoDB is done by using the "**createIndex**" method.

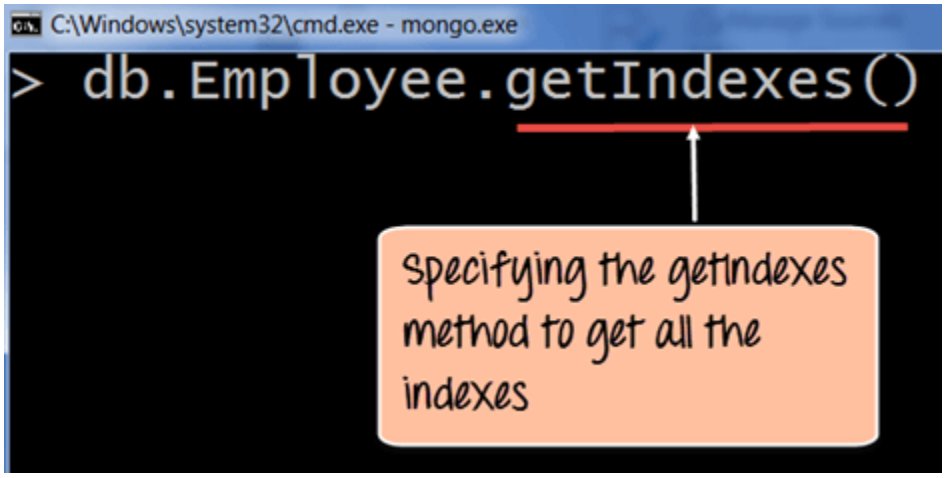
The following example shows how add index to collection. Let's assume that we have our same Employee collection which has the Field names of "Employeeid" and "EmployeeName".



How to Find Indexes: getindexes()

Finding an Index in MongoDB is done by using the **"getIndexes"** method.

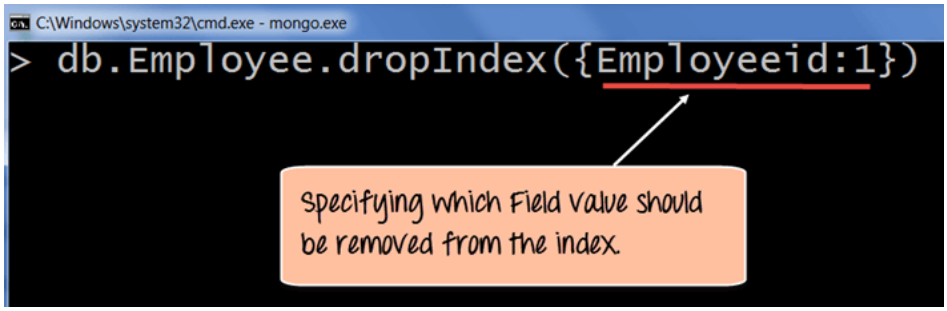
The following example shows how this can be done;



How to Drop Indexes: dropindex()

Removing an Index in MongoDB is done by using the dropIndex method.

The following example shows how this can be done;



Summary

- Defining indexes are important for faster and efficient searching of documents in a collection.

- Indexes can be created by using the createIndex method. Indexes can be created on just one field or multiple field values.
- Indexes can be found by using the getIndexes method.
- Indexes can be removed by using the dropIndex for single indexes or dropIndexes for dropping all indexes.