WEBISOFTECH

Chapter 4
Node Js MongoDB Insert Records

Node.js MongoDB Insert Record

The insertOne method is used to insert record in MongoDB's collection. The first argument of the insertOne method is an object which contains the name and value of each field in the record you want to insert.

Example

Insert Single record

Insert a record in "employees" collection.

Create a js file named "insert.js", having the following code:

```
var MongoClient = require('mongodb').MongoClient;
var url = "mongodb://localhost:27017/ MongoDatabase";
MongoClient.connect(url, function(err, db) {
if (err) throw err;
var myobj = { name: "Ajeet Kumar", age: "28", address: "Delhi" };
db.collection("employees").insertOne(myobj, function(err, res) {
if (err) throw err;
console.log("1 record inserted");
db.close();
});
});
```

Open the command terminal and run the following command:

Node insert.js

Now a record is inserted in the collection.

//..

Insert Multiple Records

You can insert multiple records in a collection by using insert() method. The insert() method uses array of objects which contain the data you want to insert.

Example

Insert multiple records in the collection named "employees".

Create a js file name insertall.js, having the following code:

```
var MongoClient = require('mongodb').MongoClient;
var url = "mongodb://localhost:27017/ MongoDatabase";
MongoClient.connect(url, function(err, db) {
if (err) throw err;
var myobj = [
{ name: "Mahesh Sharma", age: "25", address: "Ghaziabad"},
{ name: "Tom Moody", age: "31", address: "CA"},
{ name: "Zahira Wasim", age: "19", address: "Islamabad"},
{ name: "Juck Ross", age: "45", address: "London"}
];
db.collection("customers").insert(myobj, function(err, res) {
if (err) throw err;
console.log("Number of records inserted: " + res.insertedCount);
db.close();
});
});
```

Open the command terminal and run the following command:

Node insertall.js

You can see here 4 records are inserted.