

Exercise (Instructions): Node and MongoDB Part 1

Objectives and Outcomes

In this exercise you will install the Node MongoDB driver module and configure your Node application to communicate with the MongoDB server. At the end of this exercise, you will be able to:

- Install and use the Node MongoDB driver
- Interact with the MongoDB database from a Node application

Installing the Node MongoDB Driver Module

- Create a new folder named *node-mongo* and move into the folder.
- At the prompt, type the following to initialize a package.json file in the node-mongo folder:

```
1 npm init
```

- Accept the standard defaults suggested until you end up with a package.json file containing the following:


```
1 {  
2   "name": "node-mongo",  
3   "version": "1.0.0",  
4   "description": "Node MongoDB Example",  
5   "main": "index.js",  
6   "scripts": {  
7     "test": "echo \"Error: no test specified\" && exit 1",  
8     "start": "node index"  
9   },  
10  "author": "Jogesh Muppala",  
11  "license": "ISC"  
12 }
```

- Install the Node MongoDB driver and the Assert module by typing the following at the prompt:

```
1 npm install mongodb@3.0.10 --save  
2 npm install assert@1.4.1 --save
```

A Simple Node-MongoDB Application

- Create a new file named *index.js* and add the following code to it:



```

1  const MongoClient = require('mongodb').MongoClient;
2  const assert = require('assert');
3
4  const url = 'mongodb://localhost:27017/';
5  const dbname = 'conFusion';
6
7  MongoClient.connect(url, (err, client) => {
8
9      assert.equal(err, null);
10
11      console.log('Connected correctly to server');
12
13      const db = client.db(dbname);
14      const collection = db.collection("dishes");
15      collection.insertOne({ "name": "Uthappizza", "description": "test" },
16      (err, result) => {
17          assert.equal(err, null);
18
19          console.log("After Insert:\n");
20          console.log(result.ops);
21
22          collection.find({}).toArray((err, docs) => {
23              assert.equal(err, null);
24
25              console.log("Found:\n");
26              console.log(docs);
27
28              db.dropCollection("dishes", (err, result) => {
29                  assert.equal(err, null);
30
31                  client.close();
32              });
33          });
34      });
35  });
36  });

```

- Make sure that your MongoDB server is up and running
- Type the following at the prompt to start the server and see the result.

```
1  npm start
```

- Add a .gitignore file with the contents "node_modules" in the project folder.
- Initialize the Git repository, check in all the files and do a Git commit with the message "Node MongoDB Example 1".

Conclusions

In this exercise you installed the Node MongoDB driver and used it to communicate with the MongoDB server from a Node application.

Mark as completed

