Week 1: Introduction

What is MongoDB ?

1. Data stored in the form of JSON

2. JSON : Common data access model

3. Scaling out is easy

4. No joins, suitable for shard/fragmented data

Mongo —— app —— (web) clients

Mongo —— app —— (API)

app == Nodejs (V8 engine)

JS code controlling V8 C++ engine

Core server of MongoDB is written in C++

Mongoshell - C++ program control using V8

Nodejs driver - API to interact with MongoDb and Nodejs

Terminal:

Terminal 1 :mongod

Teriminal 2: mongo

>> help

>> show dbs

test

local

Global variable db

>> use video // if exists switch to database video else create

>> db.movies.insertOne({ JSON })

o/p

{

acknowledge: true,

insertedId: ObjectId(“”)

}

>> db.movies.find().pretty();

>> var c = db.movies.find()

>> c.hasNext()

>> c.next()

Hello World using Nodejs

var http = require(‘http’);

var server = http.createServer( (req, res) => {

response.writeHead(200, ‘content-type’ : ‘text/plain’);

response.end(‘hellow world’);

} );

server.listen(4000)

NPM:

npm install —save <package\_name>

OR

Add all packages with versions in package.json

type npm install in the terminal

MongoDB - Nodejs Driver architecture:

App communicates with Mongodb using using WIRE protocol.

BSON form

WIRE protocol: manages opening sockets , connection management for replica sets and error handling

Example ::

const MongoClient = require('mongodb').MongoClient,

assert = require('assert');

MongoClient.connect('mongodb://localhost:27017/', { useNewUrlParser: true }, (err, client) => {

assert.equal(null, err);

console.log('successfully connected to the server');

const db = client.db('video');

db.collection('movies').find({}).toArray((err, docs) => {

docs.forEach(doc => {

console.log(doc.title);

});

client.close();

});

// Declare success

console.log("Called find()");

});

"dependencies": {

"assert": "^1.4.1",

"mongodb": "^3.1.6"

}