NoSQL = Not Only SQL

Structured Query Language

Why Use Mongo? No schema

Everything has ‘\_id’ field ObjectId()

Mongo field types: Date (ISODate format), Boolean, Array

mongod.exe in bin directory

C:\data\db Mongo Daemon app is in here

In one terminal run mongod.exe (Mongo Daemon), in another run mongo.exe (Mongo Shell)

show dbs

use mongoBasics

db.post.insert({title: “horray!”}) db is our current database which is mongoBasics, post is a collection

show collections

Collections are like tables in a relational database.

Documents are like a row in a relational database, are individual records

quit()

ls(), cls

load(‘../../../../../users/samowl/documents/projects/treehouse-mongo-basics/seed.js’)

db.users.count() returns 6

db.posts.count() returns 4

db.users.find()

db.users.find()[0] first user

db.posts.find().limit(2) limit to two posts

var post = db.posts.find()[1];

post.title

var id = post.author

db.users.find(id)

post.description

db.getCollectionNames() returns users and posts

db.posts.getIndexes() shows “\_id” : 1

db.posts.createIndex({title: 1}) -1 for descending order, second parameter {} is called the projections parameter and contain options (where keys are fields to return and values are booleans)

db.posts.getIndexes() shows “\_id” : 1 and “title” : 1 as our two indexes

db.posts.dropIndex(‘\_id’)

db.posts.findOne()

db.posts.find({},{body: false, description: false})

db.posts.find({},{title: true, \_id: false})

db.posts.find({title: “Parenting 101”},{title: true, \_id: false})

db.posts.find({title: “How to workout”},{})

db.posts.find({$or: [{title: “Parenting 101”},{title: “My Awesome Recipe!”}]}, {})

Need to add make into user variable path for npm install -g mongo-hacker to work

db.users.find({},{\_id: true})

db.posts.find({author: ObjectId("5781a37483fcb9914560c9f4")})

db.posts.update({author: ObjectId("5781a37483fcb9914560c9f4")},{$set: {tags: [‘foo’, ‘bar’, ‘interesting’], title: “Im an Updated Title!”}}) second parameter of update is called the update parameter, $set operator. Returns nMatched : 1 and nModified : 1

If you try to update a field that does not exist on a document, by providing a field that is not already on the document, the field will be created with the values you provide.

Aggregation methods: to help sort, filter, and return results

Object.keys(db.posts.find()[0])

db.posts.find({},{title: true}).sort({title: 1})

db.posts.find({},{title: true}).limit(2).skip(2) skips the first 2 and lists next 2 (use limit and skip together for pagination)

Language drivers (also clients, libraries, plugins, bindings)

Sharding (horizontal scaling). Mongo can send write operations to all shards.

Relational databases can send write operations to only one database.

Master slave configuration: master accepts all write operations then spreads results to slaves