# Data Centric RAD

## Lab 5 MongoDB

### Setup

1. Open the Windows command prompt and navigate to the following folder:

C:\Program Files\MongoDB\Server\3.4\bin

1. Start the mongo daemon as follows:

mongod

1. In a different Windows command prompt window navigate to the following folder:

C:\Program Files\MongoDB\Server\3.4\bin

1. Start mongoDB as follows:

mongo

### Exercises

1. Create a mongodb database called usersdb.

use usersdb;

1. Create a collection called users which has documents with the following attributes:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **\_id** | **fname** | **surname** | **age** | **email** | **carReg** |
| 100 | John | Smith | 33 | [jsmith@gmail.com](mailto:jsmith@gmail.com) | 131-G-101 |
| 101 | Sean | Murphy | 21 | [seanmurph@yahoo.com](mailto:seanmurph@yahoo.com) | 04-WH-235 |
| 102 | Aine | Browne | 23 | [abrowne@gmail.com](mailto:abrowne@gmail.com) |  |
| 103 | Alan | Murphy | 24 | [murpha@hotmail.com](mailto:murpha@hotmail.com) | 07-RN-9988 |
| 104 | Sarah | Doyle | 23 | [sarah@gmail.com](mailto:sarah@gmail.com) | 142-G-2343 |
| 105 | Bill | Mulligan | 19 | [billy123@gmail.com](mailto:billy123@gmail.com) |  |
| 106 | Shane | Kelly | 24 | [sk998@yahoo.com](mailto:sk998@yahoo.com) |  |
| 107 | Will | Doyle | 19 | [doyler123@gmail.com](mailto:doyler123@gmail.com) | 10-G-2353 |

db.users.save({\_id:100, fname:"John", surname:"Smith", age:33, email:"jsmith@gmail.com" carReg:"1"})

db.users.save({\_id:101, fname:"Sean", surname:"Murphy", age:21, email:"seanmurph@yahoo.com", carReg:"04-WH-235"})

db.users.save({\_id:102, fname:"Aine", surname:"Browne", age:23, email:"abrowne@gmail.com"})

db.users.save({\_id:103, fname:"Alan", surname:"Murphy", age:24, email:"murpha@hotmail.com", carReg:"07-RN-9988"})

db.users.save({\_id:104, fname:"Sarah", surname:"Doyle", age:23, email:"sarah@gmail.com", carReg:"142-G-2343"})

db.users.save({\_id:105, fname:"Bill", surname:"Mulligan", age:19, email:"billy[123@gmail.com"}](mailto:123@gmail.com%22%7d))

db.users.save({\_id:106, fname:"Shane", surname:"Kelly", age:24, email:"sk998@yahoo.com"})

db.users.save({\_id:107, fname:"Will", surname:"Doyle", age:19, email:"doyler123@gmail.com", carReg:"10-G-2353"})

1. Give the command to find all documents in the users collection.

db.users.find()

1. Give the command to find all documents in the users collection where the age is 19.

db.users.find({age:19})

1. Give the command to find all documents in the users collection where the age greater than 19.

db.users.find({age:{$gt:19}})

1. Give the command to find all documents in the users collection where the age greater than 19 and the user has a car.

db.users.find({$and: [{age:{$gt:19}}, {carReg:{$exists:true}}]})

1. Give the command to find all documents in the users collection where \_id is greater than 104 and age is greater than 20.

db.users.find({\_id:{$gt:102}, age:{$gt:20}})

or

db.users.find({$and: [{\_id:{$gt:102}},{age:{$gt:20}}]})

1. Give the command to find the first document in the users collection where the user has a car.

db.users.findOne({carReg: {$exists:true}})

1. User 106 - Shane has bought a car with reg 12-G-1234. The following command was run to update the user’s document:

db.users.save({\_id:106, carReg:"12-G-1234"})

What does the document look like now and why?

The document looks like:

{ "\_id" : 106, "carReg" : "12-G-1234" }

Because save() takes as a parameter, the document to update. If the document \_id exists, the existing document is replaced. If the document \_id does not exist, a new document is created.

1. User 102 - Aine has bought a car with reg 10-G-9876. The following command was run to update the user’s document:

db.users.update({\_id:102}, {carReg:"10-G-9876"})

What does the document look like now and why?

The document looks like:

{ "\_id" : 102, "carReg" : "10-G-9876" }

Because update() takes as its second parameter, the document to update.

The document to update replaces the existing document if the query is found, if the query is not found, nothing is updated.

1. User 105 – Bill’s document is as follows:

{ "\_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 19, "email" : "billy123@gmail.com" }

Bill has bought a car with reg 161-MO-4. Give the command so that Bill’s document now looks as follows:

{ "\_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 19, "email" : "billy123@gmail.com", "carReg" : "161-MO-4" }

db.users.update({\_id:105},{$set:{carReg:"161-MO-4"}})

1. User 106’s document now looks as follows:

{ "\_id" : 106, "carReg" : "12-G-1234" }

Give the command to add the original fields:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **\_id** | **fname** | **surname** | **Age** | **email** |
| 106 | Shane | Kelly | 24 | [sk998@yahoo.com](mailto:sk998@yahoo.com) |

back to the document in a single command so that the document now looks as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **\_id** | **fname** | **surname** | **Age** | **email** | **carReg** |
| 106 | Shane | Kelly | 24 | [sk998@yahoo.com](mailto:sk998@yahoo.com) | 12-G-1234 |

db.users.update({\_id:106},{$set:{fname:"Shane",surname:"Kelly",age:24,email:"sk998@yahoo.com"}})

1. Give the mongodb command to add 1 to each user’s age.

db.users.update({},{$inc:{age:1}},{multi:true})

1. Add a new attribute *sex*, to each document as follows:

|  |  |
| --- | --- |
| **\_id** | **Sex** |
| 100 | M |
| 101 | M |
| 103 | M |
| 104 | F |
| 105 | M |
| 106 | M |
| 107 | M |

db.users.update({},{$set:{sex:"M"}},{multi:true})

db.users.update({\_id:104},{$set:{sex:"F"}})

*or*

db.users.update({\_id:{$ne:104}},{$set:{sex:"M"}},{multi:true})

db.users.update({\_id:104},{$set:{sex:"F"}})

1. Add a new attribute *title* with the value *Mr.*, to each document where the *sex* is M, and the age is greater than 20:

db.users.update({$and:[{age:{$gt:20}},{sex:"M"}]},{$set:{title:"Mr"}},{multi:true})

1. Users 101 – Sean, 103 – Alan and 107 – Will have sold their cars, update the collection to remove the *carReg* attribute from these documents.

db.users.update({$or:[{\_id:{$eq:101}},{\_id:{$eq:103}},{\_id:{$eq:107}}]}, {$unset:{carReg:1}}, {multi:true})

1. Give the mongodb command to list/show **only** the *fname*, *surname*, *age* and *sex* attributes of documents where the \_id is between 101 and 107 inclusive.

db.users.find({$and:[{\_id:{$gt:100}},{\_id:{$lt:108}}]}, {fname:1,surname:1,age:1,sex:1})

注意：db.users.find({1-所需要找的条件},{2-最后输出的要求})

1和2的位置不能调换!