# Data Centric RAD

## Lab 6 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 collection called users2 with the following data:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **\_id** | **fname** | **surname** | **age** | **email** | **sex** | **title** | **car** |
| 100 | John | Smith | 35 | jsmith@gmail.com | M | Mr | Reg: 131-G-101  Fuel:petrol |
| 101 | Sean | Murphy | 22 | [seanmurph@yahoo.com](mailto:seanmurph@yahoo.com) | M | Mr | Reg: 172-G-200  Electric:true |
| 102 | Aine | Browne | 23 | abrowne@gmail.com | F | Ms |  |
| 103 | Alan | Murphy | 25 | [murpha@hotmail.com](mailto:murpha@hotmail.com) | M |  |  |
| 104 | Sarah | Doyle | 24 | sarah@gmail.com | F |  | Reg:141-MO-123  Fuel:diesel |
| 105 | Bill | Mulligan | 20 | [billy123@gmail.com](mailto:billy123@gmail.com) | M |  | Reg: 12-RN-445  Fuel:Petrol |
| 106 | Shane | Kelly | 25 | sk998@yahoo.com | M | Mr | Reg: 11-WH-7783  Fuel:petrol |
| 107 | Will | Doyle | 20 | doyler123@yahoo.com | M | Mr |  |

db.user2.save({\_id:100,fname:"John",surname:"Smith",age:35,email:"jsmith@gmail.com",sex:"M",title:"Mr",car:{Reg:"131-G-101",Fuel:"petrol"}},{\_id:101,fname:"Sean",surname:"Murphy",age:"22",email:"seanmurph@yahoo.com",sex:"M",title:"Mr",car:{Reg:"172-G-200",Electric:"true"}},{\_id:102,fname:"Anie",surname:"Brown",age:23,email:"abrowne@gmail",sex:"F",title:"Ms"},{\_id:103,fname:"Alan",surname:"Murphy",age:25,sex:"M"},{\_id:104,fname:"Sarah",surname:"Doyle",age:"24",email:"sarah@gmail.com",sex:"F",car:{Reg:"141-MO-123",Fuel:"diesel"}},{\_id:105,fname:"Bill",surname:"Mulligan",age:"20",email:"billy123@gmail.com",sex:"M",car:{Reg:"12-RN-445",Fuel:"Perol"}},{\_id:106,fname:"Shane",surname:"Kelly",age:"25",email:"sk998@yahoo.com",sex:"M",title:"Mr",car:{Reg:"11-WH-7783",Fuel:"diesel"}},{\_id:107,fname:"Will",surname:"Doyle",age:"20",email:"doyler123@yahoo.com",sex:"M",title:"Mr"})

1. Give the mongodb command to show the indexes on the users2 collection.

db.user2.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_",

"ns" : "user2.user2"

}

]

1. Give the mongodb command to create an ascending index on the *age* attribute.

db.user2.createIndex({age:1})

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"ok" : 1

}

db.user2.dropIndex({age:1})

{ "nIndexesWas" : 2, "ok" : 1 }

1. Give the mongodb command to create a descending index on the *surname* attribute.

db.user2.createIndex({surname:-1})

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 2,

"numIndexesAfter" : 3,

"ok" : 1

}

> db.user2.dropIndex({surname:-1})

{ "nIndexesWas" : 3, "ok" : 1 }

1. Give the mongodb command to create a descending index on the *age* attribute.

db.user2.createIndex({age:-1})

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 2,

"numIndexesAfter" : 3,

"ok" : 1

}

> db.user2.dropIndex({age:-1})

{ "nIndexesWas" : 3, "ok" : 1 }

1. Give the mongodb command to remove the ascending index on the *age* attribute.

（将ascending的age移动到index中）

db.user2.dropIndex({age:1})

{ "nIndexesWas" : 2, "ok" : 1 }

> db.user2.getIndexes()

[

{

"v" : 2,

"key" : {

"\_id" : 1

},

"name" : "\_id\_",

"ns" : "user2.user2"

}

]

1. Give the mongodb command to show **only** the *fname* and *surname* attributes of users who don’t have a car.

db.user2.find({car:{$exists:false}},{fname:1,surname:1,\_id:0})

{"fname" : "Anie", "surname" : "Brown" }

{ "fname" : "Alan", "surname" : "Murphy" }

{ "fname" : "Will", "surname" : "Doyle" }

1. Give the mongodb command to show **only** the *fname*, *surname, reg* and *fuel* attributes of users who have a car.

db.user2.find({car:{$exists:true}},{fname:1,surname:1,"car.Reg":1,"car.Fuel":1,\_id:0})

{ "fname" : "John", "surname" : "Smith", "car" : { "Reg" : "131-G-101", "Fuel" : "petrol" } }

{ "fname" : "Sean", "surname" : "Murphy", "car" : { "Reg" : "172-G-200" } }

{ "fname" : "Sarah", "surname" : "Doyle", "car" : { "Reg" : "141-MO-123", "Fuel" : "diesel" } }

{ "fname" : "Bill", "surname" : "Mulligan", "car" : { "Reg" : "12-RN-445", "Fuel" : "Perol" } }

{ "fname" : "Shane", "surname" : "Kelly", "car" : { "Reg" : "11-WH-7783", "Fuel" : "diesel" } }

1. Give the mongodb command to list details of all users (except for their car). The list should be in sorted by name.

db.user2.find({},{car:0}).sort({fname:1})

{ "\_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 25, "sex" : "M" }

{ "\_id" : 102, "fname" : "Anie", "surname" : "Brown", "age" : 23, "email" : "abrowne@gmail", "sex" : "F", "title" : "Ms" }

{ "\_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : "20", "email" : "billy123@gmail.com", "sex" : "M" }

{ "\_id" : 100, "fname" : "John", "surname" : "Smith", "age" : 35, "email" : "jsmith@gmail.com", "sex" : "M", "title" : "Mr" }

{ "\_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : "24", "email" : "sarah@gmail.com", "sex" : "F" }

{ "\_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : "22", "email" : "seanmurph@yahoo.com", "sex" : "M", "title" : "Mr" }

{ "\_id" : 106, "fname" : "Shane", "surname" : "Kelly", "age" : "25", "email" : "sk998@yahoo.com", "sex" : "M", "title" : "Mr" }

{ "\_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : "20", "email" : "doyler123@yahoo.com", "sex" : "M", "title" : "Mr" }

1. Give the mongodb command to list **only** details of all cars. The list should be in sorted by car registration.

db.user2.find({},{car:1,\_id:0}).sort({"car.Reg":1})

{ }

{ }

{ }

{ "car" : { "Reg" : "11-WH-7783", "Fuel" : "diesel" } }

{ "car" : { "Reg" : "12-RN-445", "Fuel" : "Perol" } }

{ "car" : { "Reg" : "131-G-101", "Fuel" : "petrol" } }

{ "car" : { "Reg" : "141-MO-123", "Fuel" : "diesel" } }

{ "car" : { "Reg" : "172-G-200", "Electric" : "true" } }

1. Give the mongodb command to list the documents that have an *electric* attribute in the *car* attribute.

db.user2.find({"car.Electric":{$exists:true}})

{ "\_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : "22", "email" : "seanmurph@yahoo.com", "sex" : "M", "title" : "Mr", "car" : { "Reg" : "172-G-200", "Electric" : "true" } }

1. Give the mongodb command to list only the *fname* and *surname* attributes of users who have a *petrol car.*

db.user2.find({"car.Fuel":{$exists:true}},{fname:1,surname:1,\_id:0})

{ "fname" : "John", "surname" : "Smith" }

{ "fname" : "Sarah", "surname" : "Doyle" }

{ "fname" : "Bill", "surname" : "Mulligan" }

{ "fname" : "Shane", "surname" : "Kelly" }

1. Give the mongodb command to list/show only the *fname* and *surname* attributes of users who have a petrol car.

db.user2.find({"car.Fuel":{$exists:true}},{fname:1,surname:1,\_id:0})

{ "fname" : "John", "surname" : "Smith" }

{ "fname" : "Sarah", "surname" : "Doyle" }

{ "fname" : "Bill", "surname" : "Mulligan" }

{ "fname" : "Shane", "surname" : "Kelly" }

1. Give the mongodb command to delete all females between the ages of 21 and 24 inclusive.

db.user2.deleteMany({$and: [{sex:"F"},{age:{$gt:20}},{age:{$lt:25}}]})

{ "acknowledged" : true, "deletedCount" : 1 }

1. Give the mongodb command to delete the *fuel* attribute from all documents.

db.user2.deleteMany({"car.Fuel":{$exists:true}})

{ "acknowledged" : true, "deletedCount" : 4 }

1. Give the mongodb command to give everyone who owns a car the title of *Owner*.

db.user2.update({car:{$exists:true}},{$set:{title:"Owner"}},{multi:true})

1. Give the mongodb command to increase the *age* of all Males who own a car by 10 years.

db.user2.update({$and:[{car:{$exists:true}},{sex:"M"}]},{$inc:{age:10}},{multi:true})