

DATA ANALYTICS REFERENCE DOCUMENT	
Document Title:	Mongodb exercise sheet
Document No.:	1552511517
Author(s):	Gerhard van der Linde
Contributor(s):	

**REVISION HISTORY**

Revision	Details of Modification(s)	Reason for modification	Date	By
0	Draft release	Mongodb exercise sheet	2019/03/13 21:11	Gerhard van der Linde

## Applied Databases - Topic 6

### 1. Create a mongodb database called usersdb

```
use usersdb
```

### 2. 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	131-G-101
101	Sean	Murphy	21	seanmurph@yahoo.com	04-WH-235
102	Aine	Browne	23	abrowne@gmail.com	
103	Alan	Murphy	24	murpha@hotmail.com	07-RN-9988
104	Sarah	Doyle	23	sarah@gmail.com	142-G-2343
105	Bill	Mulligan	19	billy123@gmail.com	
106	Shane	Kelly	24	sk998@yahoo.com	
107	Will	Doyle	19	doyler123@gmail.com	10-G-2353

users.csv

```
_id,fname,surname,age,email,carReg
100,John,Smith,33,jsmith@gmail.com, 131-G-101
101,Sean,Murphy,21,seanmurph@yahoo.com, 04-WH-235
102,Aine,Browne,23,abrowne@gmail.com,
103,Alan,Murphy,24,murpha@hotmail.com, 07-RN-9988
104,Sarah,Doyle,23,sarah@gmail.com, 142-G-2343
105,Bill,Mulligan,19,billy123@gmail.com,
106,Shane,Kelly,24,sk998@yahoo.com,
```

107,Will,Doyle,19,doyler123@gmail.com, 10-G-2353

```
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
> db.user.save({_id:100,fname:,surname:,age:,email:,carReg: })
2019-03-13T21:40:32.298+0000 E QUERY    [js] SyntaxError: expected expression, got ',',
@({shell}):1:28
>
db.user.save({_id:100,fname:"John",surname:"smith",age:"33",email:"jsmith@gmail.com",carReg:"13
1-G-101"})
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 100 })
> db.user.find()
{ "_id" : 100, "fname" : "John", "surname" : "smith", "age" : "33", "email" :
"jsmith@gmail.com", "carReg" : "131-G-101" }
> db.user.find().pretty()
{
  "_id" : 100,
  "fname" : "John",
  "surname" : "smith",
  "age" : "33",
  "email" : "jsmith@gmail.com",
  "carReg" : "131-G-101"
}
> db.user.find()
{ "_id" : 100, "fname" : "John", "surname" : "smith", "age" : "33", "email" :
"jsmith@gmail.com", "carReg" : "131-G-101" }
>
db.user.save({_id:101,fname:"Sean",surname:"Murphy",age:"21",email:"seanmurphy@yahoo.com",carRe
g:"04-WH-235"})
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 101 })
> db.user.find()
{ "_id" : 100, "fname" : "John", "surname" : "smith", "age" : "33", "email" :
"jsmith@gmail.com", "carReg" : "131-G-101" }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : "21", "email" :
"seanmurphy@yahoo.com", "carReg" : "04-WH-235" }
> db
userdb
```

```
mongoimport --db userdb --collection user --type csv --headerline --file
C:\Users\121988\Documents\users.csv
2019-03-13T22:02:11.355+0000    connected to: localhost
2019-03-13T22:02:11.359+0000    imported 6 documents
```

### 3. Give the command to find all documents in the users collection.

```
> db
userdb
> db.user.find()
{ "_id" : 100, "fname" : "John", "surname" : "smith", "age" : "33", "email" :
"jsmith@gmail.com", "carReg" : "131-G-101" }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : "21", "email" :
"seanmurphy@yahoo.com", "carReg" : "04-WH-235" }
{ "_id" : 102, "fname" : "Aine", "surname" : "Browne", "age" : 23, "email" :
```

```
"abrowne@gmail.com", "carReg" : "" }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 23, "email" : "sarah@gmail.com",
"carReg" : "142-G-2343" }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 24, "email" :
"murpha@hotmail.com", "carReg" : "07-RN-9988" }
{ "_id" : 106, "fname" : "Shane", "surname" : "Kelly", "age" : 24, "email" : "sk998@yahoo.com",
"carReg" : "" }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 19, "email" :
"billy123@gmail.com", "carReg" : "" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 19, "email" :
"doyler123@gmail.com", "carReg" : "10-G-2353" }
>
```

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

```
> db.user.find({age:19})
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 19, "email" :
"billy123@gmail.com", "carReg" : "" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 19, "email" :
"doyler123@gmail.com", "carReg" : "10-G-2353" }
>
```

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

```
> db.user.find({age:{ $gt:19}})
{ "_id" : 102, "fname" : "Aine", "surname" : "Browne", "age" : 23, "email" :
"abrowne@gmail.com", "carReg" : "" }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 23, "email" : "sarah@gmail.com",
"carReg" : "142-G-2343" }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 24, "email" :
"murpha@hotmail.com", "carReg" : "07-RN-9988" }
{ "_id" : 106, "fname" : "Shane", "surname" : "Kelly", "age" : 24, "email" : "sk998@yahoo.com",
"carReg" : "" }
>
```

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

```
> db.user.find({$and: [{age:{ $gt:19}}, {carReg: { $exists:true}}]})
{ "_id" : 102, "fname" : "Aine", "surname" : "Browne", "age" : 23, "email" :
"abrowne@gmail.com", "carReg" : "" }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 23, "email" : "sarah@gmail.com",
"carReg" : "142-G-2343" }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 24, "email" :
"murpha@hotmail.com", "carReg" : "07-RN-9988" }
{ "_id" : 106, "fname" : "Shane", "surname" : "Kelly", "age" : 24, "email" : "sk998@yahoo.com",
"carReg" : "" }
```

>

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

```
> db.user.find({$and: [{_id:{$gt:104}}, {age: {$gt:20}}]}).pretty()
{
  "_id" : 106,
  "fname" : "Shane",
  "surname" : "Kelly",
  "age" : 24,
  "email" : "sk998@yahoo.com",
  "carReg" : ""
}
```

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

```
> db.user.findOne({carReg:{$exists:true}})
{
  "_id" : 100,
  "fname" : "John",
  "surname" : "smith",
  "age" : "33",
  "email" : "jsmith@gmail.com",
  "carReg" : "131-G-101"
}
```

## 9. 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?](#)<sup>1)</sup>

```
> db.users.save( {_id:106, carReg:"12-G-1234"} )
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.users.find()
{ "_id" : 100, "fname" : "John", "surname" : "Smith", "age" : 33, "email" : "jsmith@gmail.com", "carReg" : "131-G-101" }
{ "_id" : 102, "fname" : "Aine", "surname" : "Browne", "age" : 23, "email" : "abrowne@gmail.com", "carReg" : "" }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : 21, "email" : "seanmurph@yahoo.com", "carReg" : "04-WH-235" }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 24, "email" : "murpha@hotmail.com", "carReg" : "07-RN-9988" }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 23, "email" : "sarah@gmail.com",
```

```
"carReg" : "142-G-2343" }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 19, "email" :
"billy123@gmail.com", "carReg" : "" }
{ "_id" : 106, "carReg" : "12-G-1234" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 19, "email" :
"doyler123@gmail.com", "carReg" : "10-G-2353" }
>
```

## 10. 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?

```
> db.users.update({_id:102}, {carReg:"10-G-9876"})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.users.find()
{ "_id" : 100, "fname" : "John", "surname" : "Smith", "age" : 33, "email" : "jsmith@gmail.com",
"carReg" : "131-G-101" }
{ "_id" : 102, "carReg" : "10-G-9876" }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : 21, "email" :
"seanmurph@yahoo.com", "carReg" : "04-WH-235" }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 24, "email" :
"murpha@hotmail.com", "carReg" : "07-RN-9988" }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 23, "email" : "sarah@gmail.com",
"carReg" : "142-G-2343" }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 19, "email" :
"billy123@gmail.com", "carReg" : "" }
{ "_id" : 106, "carReg" : "12-G-1234" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 19, "email" :
"doyler123@gmail.com", "carReg" : "10-G-2353" }
```



Why.....

## 11. 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" }})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```



The key difference is \$set in the update command used to achieve this result.

## 12. 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

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	12-G-1234

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

## 13. Give the mongodb command to add 1 to each user's age.

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

See update <sup>2)</sup> documentation.

db.collection.update(query, update, options)

```
db.collection.update(  
  <query>,  
  <update>,  
  {  
    upsert: <boolean>,  
    multi: <boolean>,  
    writeConcern: <document>,  
    collation: <document>,  
    arrayFilters: [ <filterdocument1>, ... ]  
  }  
)
```

## 14. Add a new attribute sex, to each document as follows:

_id	Sex
100	M
101	M
103	M
104	F
105	M

_id	Sex
106	M
107	M

sex.csv

```
_id,Sex,
100,M
101,M
103,M
104,F
105,M
106,M
107,M
```

```
mongoimport --db userdb --collection users --type csv --headerline --mode merge --file
C:\Users\121988\Documents\sex.csv
2019-03-14T15:05:16.974+0000    connected to: localhost
2019-03-14T15:05:16.992+0000    error inserting documents: Performing an update on the path
'_id' would modify the immutable field '_id'
2019-03-14T15:05:16.993+0000    imported 7 documents
```



Note the addition flag used: **-mode merge**

## 15. 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({$and: [{Sex:"M"},{age:{$gt:20}}]},{$set:{title:"Mr."}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.users.update({$and: [{Sex:"M"},{age:{$gt:20}}]},{$set:{title:"Mr."},{multi:true}})
WriteResult({ "nMatched" : 4, "nUpserted" : 0, "nModified" : 3 })
> db.users.find()
{ "_id" : 100, "fname" : "John", "surname" : "Smith", "age" : 37, "email" : "jsmith@gmail.com",
  "carReg" : "131-G-101", "Sex" : "M", "title" : "Mr." }
{ "_id" : 102, "carReg" : "10-G-9876", "age" : 3 }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : 24, "email" :
  "seanmurph@yahoo.com", "carReg" : "04-WH-235", "Sex" : "M", "title" : "Mr." }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 27, "email" :
  "murpha@hotmail.com", "carReg" : "07-RN-9988", "Sex" : "M", "title" : "Mr." }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 26, "email" : "sarah@gmail.com",
  "carReg" : "142-G-2343", "Sex" : "F" }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 22, "email" :
  "billy123@gmail.com", "carReg" : "161-M0-4", "Sex" : "M", "title" : "Mr." }
{ "_id" : 106, "carReg" : "12-G-1234", "age" : 25, "email" : "sk998@yahoo.com", "fname" :
  "Shane", "surname" : "Kelly" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 20, "email" :
  "doyler123@gmail.com", "carReg" : "10-G-2353", "Sex" : "M" }
>
```

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

### Clearing the field only

```
> db.users.update({_id:{$in:[101,103,107]}},{ $set:{carReg:""}},{multi:true})
```

```
WriteResult({ "nMatched" : 3, "nUpserted" : 0, "nModified" : 3 })
```

```
> db.users.find()
{ "_id" : 100, "fname" : "John", "surname" : "Smith", "age" : 37, "email" : "jsmith@gmail.com",
"carReg" : "131-G-101", "Sex" : "M", "title" : "Mr." }
{ "_id" : 102, "carReg" : "10-G-9876", "age" : 3 }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : 24, "email" :
"seanmurph@yahoo.com", "carReg" : "", "Sex" : "M", "title" : "Mr." }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 27, "email" :
"murpha@hotmail.com", "carReg" : "", "Sex" : "M", "title" : "Mr." }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 26, "email" : "sarah@gmail.com",
"carReg" : "142-G-2343", "Sex" : "F" }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 22, "email" :
"billy123@gmail.com", "carReg" : "161-M0-4", "Sex" : "M", "title" : "Mr." }
{ "_id" : 106, "carReg" : "12-G-1234", "age" : 25, "email" : "sk998@yahoo.com", "fname" :
"Shane", "surname" : "Kelly" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 20, "email" :
"doyler123@gmail.com", "carReg" : "", "Sex" : "M" }
>
```

### Removing the field completely

```
> db.users.update({_id:{$in:[101,103,107]}},{ $unset:{carReg:1}},{multi:true})
```

```
WriteResult({ "nMatched" : 3, "nUpserted" : 0, "nModified" : 3 })
```

```
> db.users.find()
{ "_id" : 100, "fname" : "John", "surname" : "Smith", "age" : 37, "email" : "jsmith@gmail.com",
"carReg" : "131-G-101", "Sex" : "M", "title" : "Mr." }
{ "_id" : 102, "carReg" : "10-G-9876", "age" : 3 }
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : 24, "email" :
"seanmurph@yahoo.com", "Sex" : "M", "title" : "Mr." }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 27, "email" :
"murpha@hotmail.com", "Sex" : "M", "title" : "Mr." }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 26, "email" : "sarah@gmail.com",
"carReg" : "142-G-2343", "Sex" : "F" }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 22, "email" :
"billy123@gmail.com", "carReg" : "161-M0-4", "Sex" : "M", "title" : "Mr." }
{ "_id" : 106, "carReg" : "12-G-1234", "age" : 25, "email" : "sk998@yahoo.com", "fname" :
"Shane", "surname" : "Kelly" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 20, "email" :
"doyler123@gmail.com", "Sex" : "M" }
>
```



## 17. 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({_id:{$gte:101,$lte:107}},{fname:1,surname:1,age:1,sex:1})
{ "_id" : 101, "fname" : "Sean", "surname" : "Murphy", "age" : 24 }
{ "_id" : 102, "age" : 3 }
{ "_id" : 103, "fname" : "Alan", "surname" : "Murphy", "age" : 27 }
{ "_id" : 104, "fname" : "Sarah", "surname" : "Doyle", "age" : 26 }
{ "_id" : 105, "fname" : "Bill", "surname" : "Mulligan", "age" : 22 }
{ "_id" : 106, "age" : 25, "fname" : "Shane", "surname" : "Kelly" }
{ "_id" : 107, "fname" : "Will", "surname" : "Doyle", "age" : 20 }
```

## 18. Give the mongodb command to rename the fname attribute to Name.

```
> db.users.updateMany({},{$rename:{'fname':'Name'}})
{ "acknowledged" : true, "matchedCount" : 8, "modifiedCount" : 7 }
> db.users.find()
{ "_id" : 100, "surname" : "Smith", "age" : 37, "email" : "jsmith@gmail.com", "carReg" : "131-G-101", "Sex" : "M", "title" : "Mr.", "Name" : "John" }
{ "_id" : 102, "carReg" : "10-G-9876", "age" : 3 }
{ "_id" : 101, "surname" : "Murphy", "age" : 24, "email" : "seanmurph@yahoo.com", "Sex" : "M", "title" : "Mr.", "Name" : "Sean" }
{ "_id" : 103, "surname" : "Murphy", "age" : 27, "email" : "murpha@hotmail.com", "Sex" : "M", "title" : "Mr.", "Name" : "Alan" }
{ "_id" : 104, "surname" : "Doyle", "age" : 26, "email" : "sarah@gmail.com", "carReg" : "142-G-2343", "Sex" : "F", "Name" : "Sarah" }
{ "_id" : 105, "surname" : "Mulligan", "age" : 22, "email" : "billy123@gmail.com", "carReg" : "161-M0-4", "Sex" : "M", "title" : "Mr.", "Name" : "Bill" }
{ "_id" : 106, "carReg" : "12-G-1234", "age" : 25, "email" : "sk998@yahoo.com", "surname" : "Kelly", "Name" : "Shane" }
{ "_id" : 107, "surname" : "Doyle", "age" : 20, "email" : "doyler123@gmail.com", "Sex" : "M", "Name" : "Will" }
```

## 19. Export the userdb.users to a json file

```
mongoexport.exe /db:userdb /collection:users /jsonArray /pretty /out:users.json
2019-03-15T14:51:28.701+0000    connected to: localhost
2019-03-15T14:51:28.704+0000    exported 8 records
```

users.json

```
[{
  "_id": 100,
  "name": "John",
  "surname": "Smith",
  "sex": "M",
  "age": 37.0,
  "email": "jsmith@gmail.com",
  "carReg": "131-G-101",
  "titel": "Mr."
},
{
  "_id": 101,
  "name": "Sean",
```

```
"surname": "Murphy",
"sex": "M",
"age": 24.0,
"email": "seanmurph@yahoo.com",
"titel": "Mr."
},
{
  "_id": 102,
  "age": 3.0,
  "carReg": "10-G-9876"
},
{
  "_id": 103,
  "name": "Alan",
  "surname": "Murphy",
  "sex": "M",
  "age": 27.0,
  "email": "murpha@hotmail.com",
  "titel": "Mr."
},
{
  "_id": 104,
  "name": "Sarah",
  "surname": "Doyle",
  "sex": "F",
  "age": 26.0,
  "email": "sarah@gmail.com",
  "carReg": "142-G-2343"
},
{
  "_id": 105,
  "name": "Bill",
  "surname": "Mulligan",
  "sex": "M",
  "age": 22.0,
  "email": "billy123@gmail.com",
  "carReg": "161-M0-4",
  "titel": "Mr."
},
{
  "_id": 106.0,
  "name": "Shane",
  "surname": "Kelly",
  "age": 25.0,
  "email": "sk998@yahoo.com",
  "carReg": "12-G-1234"
},
{
  "_id": 107,
  "name": "Will",
  "surname": "Doyle",
  "sex": "M",
  "age": 20.0,
  "email": "doyler123@gmail.com"
}]
```

- 1)  
<https://docs.mongodb.com/manual/reference/method/db.collection.save/>
- 2)  
<https://docs.mongodb.com/manual/reference/method/db.collection.update/#db.collection.update>

From:  
<http://hdip-data-analytics.com/> - **HDip Data Analytics**

Permanent link:  
<http://hdip-data-analytics.com/submissions/worksheet/databases/topic6>

Last update: **2020/06/20 14:39**

