

Advanced Database Design

Lecture 8

Gregory S. DeLozier, Ph.D.

gdelozie@kent.edu

This Week's Topics

- Mongo review
- Starting with Mongo in Python
- Implementing the Todo app with Mongo
- Midterm
- Homework

MongoDB

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```



← field: value
← field: value
← field: value
← field: value

A MongoDB document.

Mongo

```
$ mongo
```

```
> show dbs
```

```
admin    (empty)
```

```
local    0.078GB
```

```
orthovis 0.078GB
```

```
test     (empty)
```

```
> use mydata
```

Mongo

```
> db
```

```
mydata
```

```
> j = {name : "greg"}
```

```
> k = { x : 31 }
```

```
> db.stuff.insert(j)
```

```
> db.stuff.insert(k)
```

Mongo

```
> show dbs
```

```
...
```

```
> show collections
```

```
...
```

```
db.stuff.find()
```

```
{ objects in collection }
```

Mongo

```
> cursor = db.stuff.find()
```

```
> while (cursor.hasNext())  
    printjson(cursor.next())
```

```
(... list ...)
```

- or -

```
> cursor[2]           // treat as array
```

Mongo

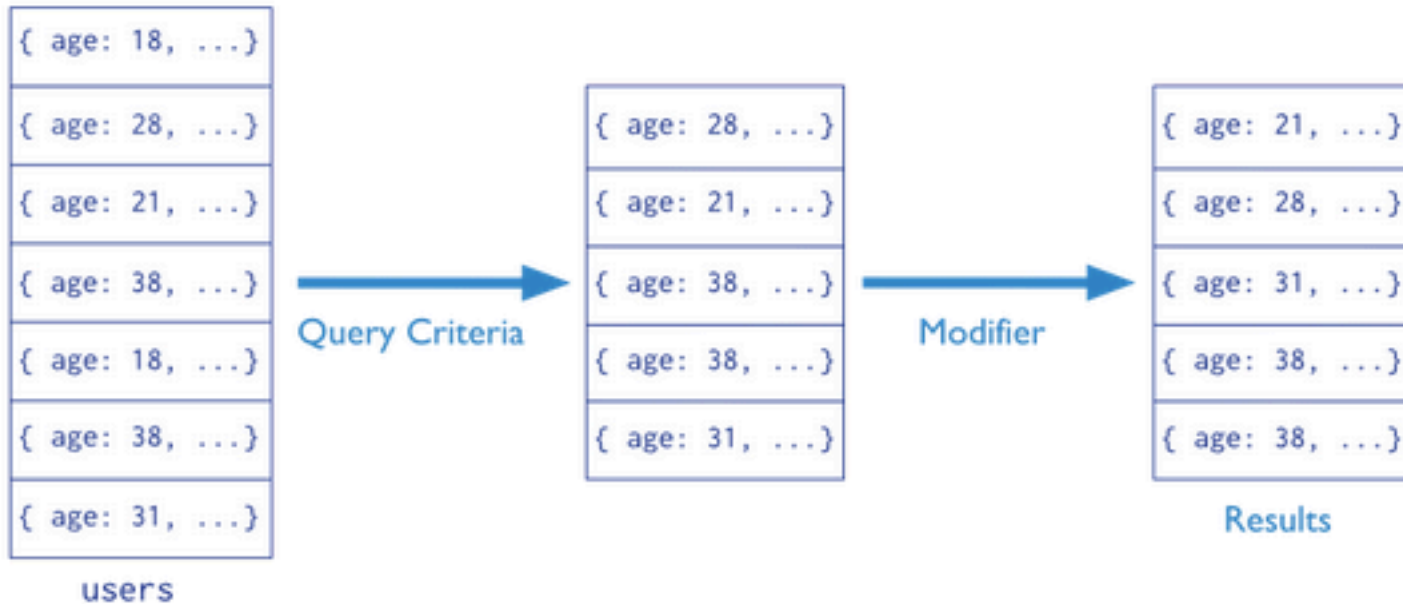
```
> cursor = db.stuff.find({age:31})
```

(or)

```
> document = db.stuff.findOne({age:31})
```


Mongo

Collection Query Criteria Modifier
`db.users.find({ age: { $gt: 18 } }).sort({age: 1 })`



The stages of a MongoDB query with a query criteria and a sort modifier.

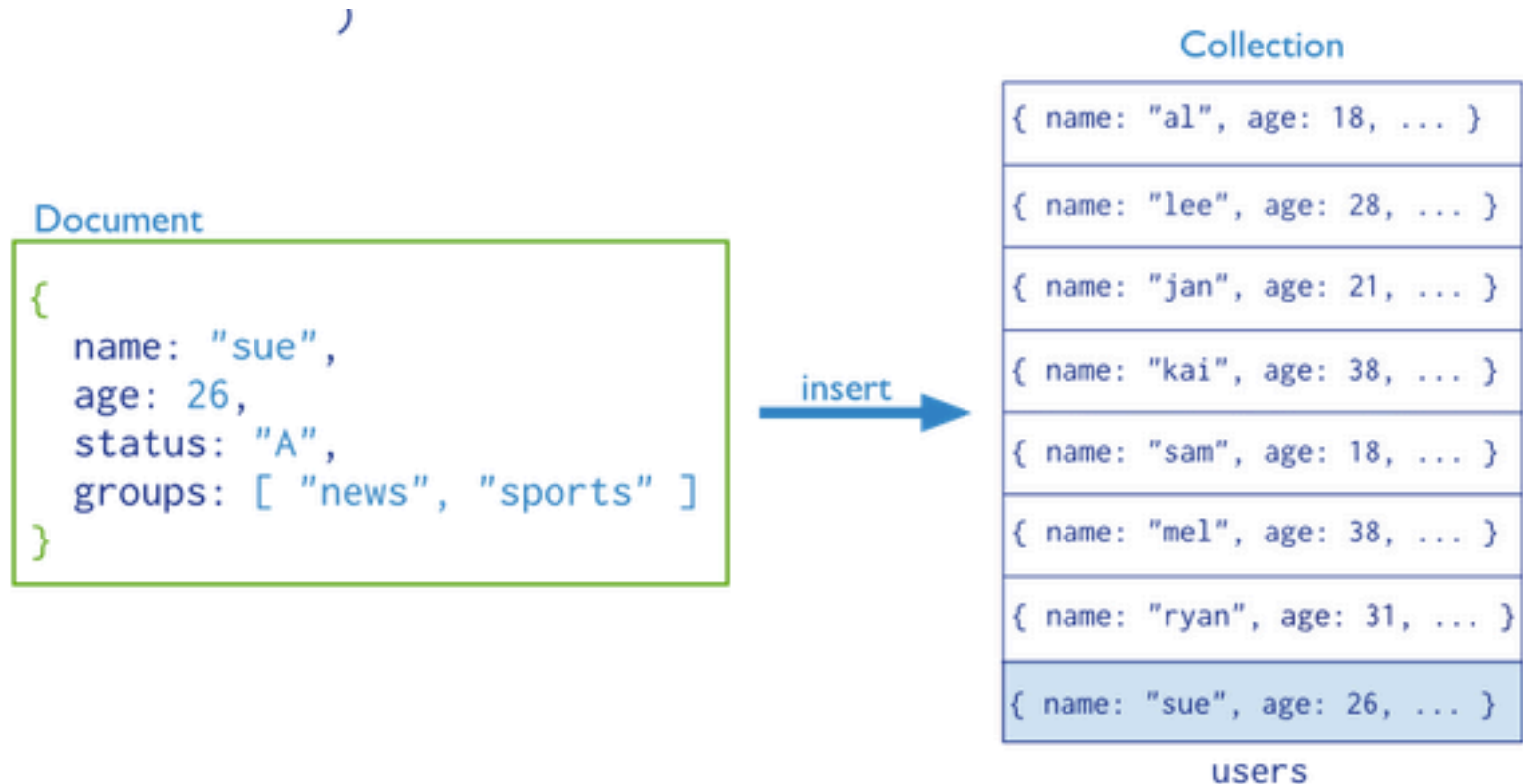
Mongo

Collection Document

↓ ↓

```
db.users.insert( {
  name: "sue",
  age: 26,
  status: "A",
  groups: [ "news", "sports" ]
})
```

Mongo



The stages of a MongoDB insert operation.

Mongo

```
> db.stuff.remove()
```

```
> db.stuff.remove({age:31})
```

```
> db.stuff.remove({age:31},1)
```

Mongo

```
db.inventory.update(  
  { item: "MNO2" },  
  {  
    $set: {  
      category: "apparel",  
      details: { model: "14Q3", manufacturer: "XYZ Company" }  
    },  
    $currentDate: { lastModified: true }  
  }  
)
```

Mongo

```
db.inventory.update(  
  { item: "ABC1" },  
  { $set: { "details.model": "14Q2" } }  
)
```

Mongo

```
db.inventory.update(  
  { category: "clothing" },  
  {  
    $set: { category: "apparel" },  
    $currentDate: { lastModified: true }  
  },  
  { multi: true }  
)
```

MongoDB

- Demo Time!

PyMongo

- Mongo client software
 - Python → pymongo
 - pymongo is *dictionary based*
 - *It's important to know how python dictionaries work*
 - Demo time!

PyMongo

```
import pymongo
```

```
client = pymongo.MongoClient()
```

```
db = client.<database_name>
```

```
db = client.stuff
```

PyMongo

```
import datetime
```

```
item = {"author": "Mike",  
        "text": "My first blog post!",  
        "tags": ["mongodb", "python", "pymongo"],  
        "date": datetime.datetime.utcnow()}
```

```
db.stuff.insert(item)
```

PyMongo

```
import datetime
```

```
item = {"author": "Mike",  
        "text": "My first blog post!",  
        "tags": ["mongodb", "python", "pymongo"],  
        "date": datetime.datetime.utcnow()}
```

```
db.stuff.insert(item)
```

```
print db.collection_names
```

PyMongo

```
db.stuff.find_one()
```

```
db.stuff.find_one({"author":"Mike"})
```

```
db.stuff.find_one({"_id":id}) #<---- not string
```

PyMongo

```
db.stuff.find ()
```

```
db.stuff.find ({"author":"Mike"})
```

```
posts.find({"date": {"$lt": d}}).sort("author")
```

PyMongo

```
db.stuff.save (<item>)
```

```
db.stuff.update ({"author":"Mike"},  
                 {"$set":{"age:44}})
```

```
db.stuff.remove({"author":"Mike"})
```

MongoDB

- Resources

- <http://www.mongodb.org>
- <http://www.mongodb.org/about/introduction/>
- <http://docs.mongodb.org/manual/>
- <https://mongolab.com/>