Taller MongoDB

Msc. Carlos Andres Lopez

Modificar

Clausula update

db.coleccion.update(criterio, dato)

Ejemplo:

```
db.libros.find({titulo:'La peste'}).pretty()
{
    "_id" : ObjectId("5d860fca968af071d31fa2b1"),
    "titulo" : "La peste",
    "autor" : "Albert Camus"
}
```

```
db.libros.update({titulo: 'La peste'},
        {$set:  { temas: [ 'literatura', 'existencialismo']}     }
    )
```

Eliminar

Clausula remove

db.coleccion.remove(criterio)

Ejemplo:

```
db.libros.findOne({titulo:'La peste'})
{
    "_id" : ObjectId("5d8612f9968af071d31fa2b3"),
    "titulo" : "La peste",
    "autor" : "Albert Camus",
    "temas" : [
        "literatura",
        "existencialismo"
    ]
}
```

db.libros.remove({titulo:'La peste'})

Buscar

Clausula find

db.coleccion.find(criterio, campos)

db.coleccion.findOne()

Ejemplo:

```
db.libros.find()
{ "_id" : ObjectId("5c968a74ceba323a3387fa8d"), "titulo" : "Codigo
Da Vinci", "autor" : "Dan Brown", "temas" : [ "arte", "historia",
"ficcion" ] }
{ "_id" : ObjectId("5c968a9fceba323a3387fa8e"), "titulo" : "La
divina comedia", "autor" : "Dante" }
{ "_id" : ObjectId("5d8612f9968af071d31fa2b3"), "titulo" : "La
peste", "autor" : "Albert Camus", "temas" : [ "literatura",
"existencialismo" ] }
```

Comparadores

Numericos

Operation	Syntax	Example	RDBMS Equivalent
Equality	{<key>:<value>}	db.mycol.find({"by":"tutorials point"}).pretty()	where by = 'tutorials point'
Less Than	{<key>:{$lt:<value>}}	db.mycol.find({"likes":{$lt:50}}).pretty()	where likes < 50
Less Than Equals	{<key>:{$lte:<value>}}	db.mycol.find({"likes":{$lte:50}}).pretty()	where likes <= 50
Greater Than	{<key>:{$gt:<value>}}	db.mycol.find({"likes":{$gt:50}}).pretty()	where likes > 50
Greater Than Equals	{<key>:{$gte:<value>}}	db.mycol.find({"likes":{$gte:50}}).pretty()	where likes >= 50
Not Equals	{<key>:{$ne:<value>}}	db.mycol.find({"likes":{$ne:50}}).pretty()	where likes != 50

Agregados (basicos)

- Contar

  db.coleccion.find(criterio).count()
- Ordenar

  db.coleccion.find(criterio).(sort{criterio:1/-1})
- Limitar

  db.coleccion.find(criterio).limit(numero)

Logicos

- And

  db.coleccion.find({criterio1, criterio2})
- Or

  db.coleccion.find({

  $or:[

  {criterio1}, {criterio2}]

  })

Agregados

Clausula aggregate permite combinar, posee tres secciones, consulta, accion, proyeccion.

```
db.orders.aggregate([
    { $match: { status: "A" } },
    { $group: { _id: null, total: { $sum: "$cantidad" } } },
    { $sort: { total: -1 } }
])
```

Agregados

Expression	Description	Example
$sum	Sums up the defined value from all documents in the collection.	db.mycol.aggregate([{$group : {_id : "$by_user", num_tutorial : {$sum : "$likes"}}}])
$avg	Calculates the average of all given values from all documents in the collection.	db.mycol.aggregate([{$group : {_id : "$by_user", num_tutorial : {$avg : "$likes"}}}])
$min	Gets the minimum of the corresponding values from all documents in the collection.	db.mycol.aggregate([{$group : {_id : "$by_user", num_tutorial : {$min : "$likes"}}}])
$max	Gets the maximum of the corresponding values from all documents in the collection.	db.mycol.aggregate([{$group : {_id : "$by_user", num_tutorial : {$max : "$likes"}}}])

Agregados

$push	Inserts the value to an array in the resulting document.	db.mycol.aggregate([{$group : {_id : "$by_user", url : {$push: "$url"}}}])
$addToSet	Inserts the value to an array in the resulting document but does not create duplicates.	db.mycol.aggregate([{$group : {_id : "$by_user", url : {$addToSet : "$url"}}}])
$first	Gets the first document from the source documents according to the grouping. Typically this makes only sense together with some previously applied "$sort"-stage.	db.mycol.aggregate([{$group : {_id : "$by_user", first_url : {$first : "$url"}}}])
$last	Gets the last document from the source documents according to the grouping. Typically this makes only sense together with some previously applied "$sort"-stage.	db.mycol.aggregate([{$group : {_id : "$by_user", last_url : {$last : "$url"}}}])

Agregados

- $project: especifica campos
- $match: especifica filtro
- $group: agrupa elementos
- $sort: permite ordenar.
- $skip: permite omitir
- $limit: establece limite
- $unwind: desagrega elementos en arreglos