

Curso Intro BigData-MongoDB

MongoDB – Consultas Parte 1



Ejecutables básicos

Servidor MongoDB

mongod

```
--dbpath      <directorio/de/datos>  
--port        <puerto>  
--config      <path/al/archivo_de_configuracion>  
--logpath     <path/al/log>  
--fork  
--auth
```

Ejecutables básicos – Archivo de configuración

<http://docs.mongodb.org/manual/reference/configuration-options/>

where to write logging data.

systemLog:

destination: file

logAppend: true

path: /var/log/mongodb/mongod.log

Where and how to store data.

storage:

dbPath: /var/lib/mongo

journal:

enabled: true

engine:

mmapv1:

wiredTiger:



Ejecutables básicos – Archivo de configuración

```
# how the process runs
```

```
processManagement:
```

```
  fork: true  # fork and run in background
```

```
  pidFilePath: /var/run/mongodb/mongod.pid  # location of pidfile
```

```
  timeZoneInfo: /usr/share/zoneinfo
```

```
# network interfaces
```

```
net:
```

```
  port: 27017
```

```
  bindIp: 127.0.0.1  # Listen to local interface only, comment to listen on  
all interfaces.
```

```
#security:
```

```
#operationProfiling:
```

```
#replication:
```

```
#sharding:
```



Ejecutables básicos

Shell MongoDB

mongo

```
--host          <ip_servidor> | <ip_servidor>:<puerto_servidor>
--port          <puerto_servidor>
--nodb          (permite abrir el shell sin conectarse a una db)
-u             <usuario>
-p             <password>
--eval          '<código_javascript_a_ejecutar>'
--authenticationDatabase <db_autenticacion>
<db>
<path/javascript_a_ejecutar>
```

Archivos de una BD – MMAPv1

```
drwxr-xr-x  4 mongodb mongodb      4096 may  9 11:05 .
drwxr-xr-x 54 root      root        4096 may  5 21:45 ..
-rw-----  1 mongodb nogroup 16777216 may  9 11:16 finanzas.0
-rw-----  1 mongodb nogroup 33554432 may  9 11:05 finanzas.1
-rw-----  1 mongodb nogroup 16777216 may  9 11:16 finanzas.ns
drwxr-xr-x  2 mongodb nogroup      4096 may  9 11:05 journal
-rwxr-xr-x  1 mongodb nogroup         4 may  9 10:43 mongod.lock
drwxr-xr-x  2 mongodb nogroup      4096 may  9 11:05 _tmp
```

Journal directory contiene el REDO log para un CRASH RECOVERY.

El archivo mongod.lock de 4 bytes indica que la base está siendo accedida, o sea representa un lock sobre la base de datos.

El archivo finanzas.ns (name space) contiene el catálogo de la BD “finanzas”.

Los archivos finanzas.0 y finanzas.1 contienen la data de la BD “finanzas”.

- Se crean automáticamente.
- No es necesario crear tablespaces como en otras bd.
- Prealocan espacio y van creciendo.
- Tienen un tope máximo de 2Gb por archivo.

Archivos de una BD – WiredTiger

```
$ ls
_mdb_catalog.wt                index-8--6965038582220197044.wt
collection-0--6965038582220197044.wt journal/
collection-2--6965038582220197044.wt mongod.lock
collection-4--6965038582220197044.wt sizeStorer.wt
collection-6--6965038582220197044.wt storage.bson
diagnostic.data/               WiredTiger
index-1--6965038582220197044.wt WiredTiger.lock
index-3--6965038582220197044.wt WiredTiger.turtle
index-5--6965038582220197044.wt WiredTiger.wt
index-7--6965038582220197044.wt WiredTigerLAS.wt
```

```
> db.getMongo().getDBNames().forEach(function(d)
){db.getSisterDB(d).getCollectionNames().forEach(
function(c){print(db.getSisterDB(d)[c])}})}
admin.system.version
config.system.sessions → 2 índices: _id y TTL
finanzas.facturas
local.startup_log
```

Operaciones sobre una Colección

Instrucción es sobre una Colección

db.coleccion.help()

```
MongoDB Enterprise > db.coleccion.help()
DBCollection help
  db.coleccion.find().help() - show DBCursor help
  db.coleccion.bulkWrite( operations, <optional params> ) - bulk execute write operations, optional parameters are: w, wtimeout,
j
  db.coleccion.count( query = {}, <optional params> ) - count the number of documents that matches the query, optional parameter
s are: limit, skip, hint, maxTimeMS
  db.coleccion.countDocuments( query = {}, <optional params> ) - count the number of documents that matches the query, optional
parameters are: limit, skip, hint, maxTimeMS
  db.coleccion.estimatedDocumentCount( <optional params> ) - estimate the document count using collection metadata, optional par
ameters are: maxTimeMS
  db.coleccion.copyTo(newColl) - duplicates collection by copying all documents to newColl; no indexes are copied.
  db.coleccion.convertToCapped(maxBytes) - calls {convertToCapped:'coleccion', size:maxBytes} command
  db.coleccion.createIndex(keypattern[,options])
  db.coleccion.createIndexes([keypatterns], <options>)
  db.coleccion.dataSize()
  db.coleccion.deleteOne( filter, <optional params> ) - delete first matching document, optional parameters are: w, wtimeout, j
  db.coleccion.deleteMany( filter, <optional params> ) - delete all matching documents, optional parameters are: w, wtimeout, j
  db.coleccion.distinct( key, query, <optional params> ) - e.g. db.coleccion.distinct( 'x' ), optional parameters are: maxTimeMS

  db.coleccion.drop() drop the collection
  db.coleccion.dropIndex(index) - e.g. db.coleccion.dropIndex( "indexName" ) or db.coleccion.dropIndex( { "indexKey" : 1 } )
  db.coleccion.dropIndexes()
  db.coleccion.ensureIndex(keypattern[,options]) - DEPRECATED, use createIndex() instead
  db.coleccion.explain().help() - show explain help
  db.coleccion.reIndex()
  db.coleccion.find([query],[fields]) - query is an optional query filter. fields is optional set of fields to return.
  e.g. db.coleccion.find( {x:77} , {name:1, x:1} )
```


Consultando una Colección

Consultar todos los documentos de una colección

```
db.facturas.find()
```

```
> db.facturas.find()
{ "_id" : ObjectId("534b52b443ad02d44397b71b"), "nroFactura" : 1448, "fechaEmisi
on" : ISODate("2014-02-20T00:00:00Z"), "fechaVencimiento" : ISODate("2014-03-22T
00:00:00Z"), "condPago" : "30 Ds FF", "cliente" : { "nombre" : "Martin", "apelli
do" : "Zavasi", "cuit" : 2038373771, "region" : "CABA" }, "item" : [ { "produc
to" : "CORREA 10mm", "cantidad" : 2, "precio" : 134 } ] }
{ "_id" : ObjectId("534b52bc43ad02d44397b71c"), "nroFactura" : 1449, "fechaEmisi
on" : ISODate("2014-02-20T00:00:00Z"), "fechaVencimiento" : ISODate("2014-02-20T
00:00:00Z"), "condPago" : "CONTADO", "cliente" : { "nombre" : "Martín", "apellid
o" : "Zavasi", "cuit" : 2038373771, "region" : "CABA" }, "item" : [ {
"producto" : "TUERCA 2mm", "cantidad" : 6, "precio" : 60 },
{ "producto" : "CORREA 10mm", "cantidad" : 12, "precio" : 134 }
] }
```

Consultando una Colección

Consultar un documento de una colección

`db.facturas.findOne()`

```
> db.facturas.findOne()
{
  "_id" : ObjectId("534b52b443ad02d44397b71b"),
  "nroFactura" : 1448,
  "fechaEmision" : ISODate("2014-02-20T00:00:00Z"),
  "fechaVencimiento" : ISODate("2014-03-22T00:00:00Z"),
  "condPago" : "30 Ds FF",
  "cliente" : {
    "nombre" : "Martin",
    "apellido" : "Zavasi",
    "cuit" : 2038373771,
    "region" : "CABA"
  },
  "item" : [
    {
      "producto" : "CORREA 10mm",
      "cantidad" : 2,
      "precio" : 134
    }
  ]
}
```

Consultando una Colección

```
> db.facturas.find().limit(2).toArray()
[
  {
    "_id" : ObjectId("534b52b443ad02d44397b71b"),
    "nroFactura" : 1448,
    "fechaEmision" : ISODate("2014-02-20T00:00:00Z"),
    "fechaVencimiento" : ISODate("2014-03-22T00:00:00Z"),
    "condPago" : "30 Ds FF",
    "cliente" : {
      "nombre" : "Martin",
      "apellido" : "Zavasi",
      "cuit" : 2038373771,
      "region" : "CABA"
    },
    "item" : [
      {
        "producto" : "CORREA 10mm",
        "cantidad" : 2,
        "precio" : 134
      }
    ]
  },
  {
    "_id" : ObjectId("534b52bc43ad02d44397b71c"),
    "nroFactura" : 1449,
    "fechaEmision" : ISODate("2014-02-20T00:00:00Z"),
    "fechaVencimiento" : ISODate("2014-02-20T00:00:00Z"),
    "condPago" : "CONTADO",
    "cliente" : {
      "nombre" : "Martín",
      "apellido" : "Zavasi",
      "cuit" : 2038373771,
      "region" : "CABA"
    },
    "item" : [
      {
        "producto" : "TUERCA 2mm",
        "cantidad" : 6,
        "precio" : 60
      },
      {
        "producto" : "CORREA 10mm",
        "cantidad" : 12,
        "precio" : 134
      }
    ]
  }
]
```

Consultar los documentos de una colección a un array

```
db.facturas.find().toArray()
```

Nota: Podría haber problemas cuando devuelve muchos valores, ya que el array se trataría de cargar todo en memoria.

Para ese caso deberíamos limitar la consulta de la siguiente manera:

```
db.facturas.find().limit(2).toArray()
```



Consultando una Colección

```
> db.facturas.find().limit(2).skip(2).pretty()
{
  "_id" : ObjectId("53685dbb2baf7b93f61df564"),
  "nroFactura" : 1447,
  "fechaEmision" : ISODate("2014-02-20T00:00:00Z"),
  "fechaVencimiento" : ISODate("2014-02-20T00:00:00Z"),
  "condPago" : "CONTADO",
  "cliente" : {
    "nombre" : "Marina",
    "apellido" : "Malinez",
    "cuit" : 2740488484,
    "region" : "CENTRO"
  },
  "item" : [
    {
      "producto" : "CORREA 12mm",
      "cantidad" : 11,
      "precio" : 18
    },
    {
      "producto" : "TALADRO 12mm",
      "cantidad" : 1,
      "precio" : 490
    }
  ]
},
{
  "_id" : ObjectId("53685dbb2baf7b93f61df565"),
  "nroFactura" : 1448,
  "fechaEmision" : ISODate("2014-02-20T00:00:00Z"),
  "fechaVencimiento" : ISODate("2014-03-22T00:00:00Z"),
  "condPago" : "30 Ds FF",
  "cliente" : {
    "nombre" : "Martin",
    "apellido" : "Zavasi",
    "cuit" : 2038373771,
    "region" : "CABA"
  },
  "item" : [
    {
      "producto" : "CORREA 10mm",
      "cantidad" : 2,
      "precio" : 134
    }
  ]
}
```

Consultar dos documentos, saltando los dos primeros documentos de una colección, mostrándolos en un modo mejorado.

```
db.facturas.find().limit(2).skip(2).pretty()
```

Paginación por consultas a la DB

```
> db.facturas.find(<>,{nroFactura:1,_id:0}).limit(2)
{ "nroFactura" : 1000 }
{ "nroFactura" : 1001 }
>
> db.facturas.find(<>,{nroFactura:1,_id:0}).skip(2).limit(2)
{ "nroFactura" : 1002 }
{ "nroFactura" : 1003 }
>
> db.facturas.find(<>,{nroFactura:1,_id:0}).skip(4).limit(2)
{ "nroFactura" : 1004 }
{ "nroFactura" : 1005 }
>
> db.facturas.find(<>,{nroFactura:1,_id:0}).skip(6).limit(2)
{ "nroFactura" : 1006 }
{ "nroFactura" : 1007 }
```

Usando las funciones `skip()` y `limit()` podríamos manejar paginación mediante consultas sucesivas a la DB. Limitando la cantidad que quiero mostrar en cada página y saltando los resultados vistos en páginas anteriores.

```
db.facturas.find().skip(p*n).limit(n)
```

Donde `n` es la cantidad de resultado que quiero por página y `p` el número de página actual.

Nota: Cabe aclarar que este tipo de paginado no es recomendable cuando la cantidad de documentos que quiero saltar es muy grande, ya que deberán encontrarse esos documentos para luego poder saltarlos y devolver los siguientes.

Consultando una Colección – Criterios de Selección

Buscar documentos cuyo condición de pago sea igual a "CONTADO",
mostrando todos sus atributos.

```
db.facturas.find( { condPago : "CONTADO" }, {})
```

```
> db.facturas.find(<condPago:"CONTADO">,<>)
{ "_id" : ObjectId("53685dbb2baf7b93f61df564"), "nroFactura" : 1447, "fechaEmisi
on" : ISODate("2014-02-20T00:00:00Z"), "fechaVencimiento" : ISODate("2014-02-20T
00:00:00Z"), "condPago" : "CONTADO", "cliente" : { "nombre" : "Marina", "apellid
o" : "Malinez", "cuit" : 2740488484, "region" : "CENTRO" }, "item" : [ {
"producto" : "CORREA 12mm", "cantidad" : 11, "precio" : 18 },
{ "producto" : "TALADRO 12mm", "cantidad" : 1, "precio" : 490 }
] }
{ "_id" : ObjectId("53685dbb2baf7b93f61df567"), "nroFactura" : 1450, "fechaEmisi
on" : ISODate("2014-02-24T00:00:00Z"), "fechaVencimiento" : ISODate("2014-02-24T
00:00:00Z"), "condPago" : "CONTADO", "cliente" : { "nombre" : "Juan Manuel", "ap
ellido" : "Manoni", "cuit" : 2029889382, "region" : "NEA" }, "item" : [
{ "producto" : "TUERCA 2mm", "cantidad" : 2, "precio" : 60 },
{ "producto" : "TALADRO 12mm", "cantidad" : 1, "precio"
: 490 },
{ "producto" : "TUERCA 5mm", "cantidad" : 15,
"precio" : 90 } ] }
{ "_id" : ObjectId("53685dbb2baf7b93f61df56a"), "nroFactura" : 1453, "fechaEmisi
```

Consultando una Colección – Criterios de Selección

Buscar documentos que contengan precios de items sean iguales a 490, mostrando sólo los atributos `_id`, `nro` de factura y array de item. El atributo `_id` lo muestra por default.

```
db.facturas.find( { "item.precio":490 } , {nroFactura:1,item:1})
```

```
>
> db.productos.find( { "item.precio":490 } , {nroFactura:1,item:1})
> db.facturas.find( { "item.precio":490 } , {nroFactura:1,item:1})
{ "_id" : ObjectId("53685dbb2baf7b93f61df564"), "nroFactura" : 1447, "item" : [
  {
    "producto" : "CORREA 12mm", "cantidad" : 11, "precio"
: 18 },
  {
    "producto" : "TALADRO 12mm", "cantidad" : 1,
"precio" : 490 } ] }
{ "_id" : ObjectId("53685dbb2baf7b93f61df567"), "nroFactura" : 1450, "item" : [
  {
    "producto" : "TUERCA 2mm", "cantidad" : 2, "precio"
: 60 },
  {
    "producto" : "TALADRO 12mm", "cantidad" : 1,
"precio" : 490 },
  {
    "producto" : "TUERCA 5mm", "cantidad" : 15,
"precio" : 90 } ] }
{ "_id" : ObjectId("53685dbb2baf7b93f61df569"), "nroFactura" : 1452, "item" : [
  {
    "producto" : "SET HERRAMIENTAS", "cantidad" : 1,
"precio" : 700 },
  {
    "producto" : "TALADRO 12mm", "cantidad" : 1,
"precio" : 490 } ] }
```

Consultando una Colección – Criterios de Selección

Buscar documentos cuyo precio sean iguales a 490, mostrando sólo los atributos nroFactura y fechaEmision. El atributo _id se pide explícitamente que no aparezca

```
db.facturas.find( { "item.precio":490 } , {_id:0,nroFactura:1,fechaEmision:1})
```

```
> db.facturas.find(<< "item.precio":490 > , <_id:0, nroFactura:1,fechaEmision:1>>)
{ "nroFactura" : 1447, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1450, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
{ "nroFactura" : 1452, "fechaEmision" : ISODate("2014-02-25T00:00:00Z") }
{ "nroFactura" : 1454, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1457, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
{ "nroFactura" : 1459, "fechaEmision" : ISODate("2014-02-25T00:00:00Z") }
{ "nroFactura" : 1461, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1464, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
{ "nroFactura" : 1466, "fechaEmision" : ISODate("2014-02-25T00:00:00Z") }
{ "nroFactura" : 1468, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1471, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
{ "nroFactura" : 1473, "fechaEmision" : ISODate("2014-02-25T00:00:00Z") }
{ "nroFactura" : 1475, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1478, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
{ "nroFactura" : 1480, "fechaEmision" : ISODate("2014-02-25T00:00:00Z") }
{ "nroFactura" : 1482, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1485, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
{ "nroFactura" : 1487, "fechaEmision" : ISODate("2014-02-25T00:00:00Z") }
{ "nroFactura" : 1489, "fechaEmision" : ISODate("2014-02-20T00:00:00Z") }
{ "nroFactura" : 1492, "fechaEmision" : ISODate("2014-02-24T00:00:00Z") }
Type "it" for more
```


Consultando una Colección – Criterios de Selección

Es importante al comparar tener en cuenta el tipo de dato:

```
db.facturas.find( { "item.precio" : 490 } )
```

no es lo mismo que

```
db.facturas.find( { "item.precio" : "490" } )
```

```
db.facturas.find( { "item.precio" : "490" } , { _id:0, nroFactura:1, fechaEmision:1 })
```

```
>
> db.facturas.find(<{ "item.precio":"490" } , { _id:0, nroFactura:1, fechaEmision:1
>
>
```

Consultando una Colección – Criterios de Selección

Buscar documentos cuya número de factura sea igual a 1450 y la condición de pago sea igual a "CONTADO".
Mostrando todos sus atributos.

```
db.facturas.find( { nroFactura : 1450, condPago:"CONTADO" } ,{})
```

```
>
> db.facturas.find( { nroFactura : 1450, condPago:"CONTADO" } ,{})
{ "_id" : ObjectId("53685dbb2baf7b93f61df567"), "nroFactura" : 1450, "fechaEmisi
on" : ISODate("2014-02-24T00:00:00Z"), "fechaUencimiento" : ISODate("2014-02-24T
00:00:00Z"), "condPago" : "CONTADO", "cliente" : { "nombre" : "Juan Manuel", "ap
ellido" : "Manoni", "cuit" : 2029889382, "region" : "NEA" }, "item" : [
{ "producto" : "TUERCA 2mm", "cantidad" : 2, "precio" : 60 },
{ "producto" : "TALADRO 12mm", "cantidad" : 1, "precio"
: 490 },
{ "producto" : "TUERCA 5mm", "cantidad" : 15,
"precio" : 90 } ] }
>
```

Consultando una Colección – Criterios de Selección

Busca la cantidad de facturas cuyo Nro. de Factura sea mayor que 1465

Operadores \$

```
db.facturas.find( { nroFactura : { $gt: 1465 } } ).count()
```

\$gt

\$gte

\$lt

\$lte

\$not

\$or

\$in

\$nin

\$exist

\$regex

```
>
> db.facturas.find( { nroFactura : { $gt: 1465 } } ).count()
30512
>
```

Busca las facturas cuya fecha de emisión sea mayor o igual al 24/02/2014.

```
db.facturas.find({ fechaEmision: { $gte: ISODate("2014-02-24T00:00:00Z") } } )
```

```
> db.facturas.find( { fechaEmision: { $gte: ISODate("2014-02-24T00:00:00Z") } } )
{ "_id" : ObjectId("53685dbb2baf7b93f61df567"), "nroFactura" : 1450, "fechaEmisi
on" : ISODate("2014-02-24T00:00:00Z"), "fechaVencimiento" : ISODate("2014-02-24T
00:00:00Z"), "condPago" : "CONTADO", "cliente" : { "nombre" : "Juan Manuel", "ap
ellido" : "Manoni", "cuit" : 2029889382, "region" : "NEA" }, "item" : [
  { "producto" : "TUERCA 2mm", "cantidad" : 2, "precio" : 60 },
  { "producto" : "TALADRO 12mm", "cantidad" : 1, "precio"
: 490 },
  { "producto" : "TUERCA 5mm", "cantidad" : 15,
"precio" : 90 } ] }
{ "_id" : ObjectId("53685dbb2baf7b93f61df568"), "nroFactura" : 1451, "fechaEmisi
on" : ISODate("2014-02-24T00:00:00Z"), "fechaVencimiento" : ISODate("2014-03-26T
00:00:00Z"), "condPago" : "30 Ds FF", "cliente" : { "nombre" : "Soledad", "apell
ido" : "Lavagno", "cuit" : 2729887543, "region" : "NOA" }, "item" : [ { "produ
```

Consultando una Colección – Criterios de Selección

Busca la cantidad de facturas que contengan ítems cuyo precio sea menor que 100.

Operadores \$

```
db.facturas.find( { "item.precio": { $lt: 100 } } ).count()
```

\$gt

\$gte

\$lt

\$lte

\$not

\$or

\$in

\$nin

\$exist

\$regex

```
>  
> db.facturas.find( { "item.precio": { $lt: 100 } } ).count()  
17737  
>
```

Busca las facturas en los que no exista el campo indicador, listando los atributos `_id`, nombre del cliente e indicador.

```
db.facturas.find( { indicador: { $exists: false } }, { "cliente.nombre": 1 } )
```

Consultando una Colección – Criterios de Selección

Muestra los primeros dos productos de la consulta, que tenga un producto de 20 pesos o menos, mostrándolo de una forma más legible.

Operadores \$

```
db.facturas.find( { "item.precio" : { $lte: 20 } } ).pretty().limit(2)
```

\$gt
\$gte
\$lt
\$lte
\$not
\$or
\$in
\$nin
\$exist
\$regex

```
> db.facturas.find( { condPago : { $lte: "30 Ds FF" } } ).pretty().limit(2)
{
  "_id" : ObjectId("544fbeca9984ce2ac8036682"),
  "nroFactura" : 1455,
  "fechaEmision" : ISODate("2014-02-20T00:00:00Z"),
  "fechaVencimiento" : ISODate("2014-03-22T00:00:00Z"),
  "condPago" : "30 Ds FF",
  "cliente" : {
    "nombre" : "Martin",
    "apellido" : "Zavasi",
    "cuit" : 2038373771,
    "region" : "CABA"
  },
  "item" : [
    {
      "producto" : "CORREA 10mm",
      "cantidad" : 2,
      "precio" : 134
    }
  ]
}
{
  "_id" : ObjectId("544fbeca9984ce2ac8036685"),
  "nroFactura" : 1458,
  "fechaEmision" : ISODate("2014-02-24T00:00:00Z"),
  "fechaVencimiento" : ISODate("2014-03-26T00:00:00Z"),
  "condPago" : "30 Ds FF",
  "cliente" : {
    "nombre" : "Soledad",
    "apellido" : "Lavagno",
    "cuit" : 2729887543,
    "region" : "NOA"
  },
  "item" : [
    {
      "producto" : "SET HERRAMIENTAS",
      "cantidad" : 1,
      "precio" : 700
    }
  ]
}
```



Consultando una Colección – Criterios de Selección

Cuenta las facturas cuyo número no sea mayor o igual 1500.

Operadores \$

```
db.facturas.find({nroFactura: {$not: {$gte:1500}}}).count()
```

*En este caso esta consulta trae todas las **facturas con número menor a 1500** y las que **no tienen el atributo nroFactura** también.*

\$gt

\$gte

\$lt

\$lte

\$not

\$or

\$in

\$nin

\$exist

\$regex

A diferencia de:

```
db.facturas.find({ nroFactura : {$lt:1500} } ).count()
```

*Esta consulta sólo traerá los documentos cuyas **números de facturas sean menores que 1500 solamente.***

```
>
> db.facturas.find(< nroFactura : < $not: <$gte:1500> > > ).count()
1395
> db.facturas.find(< nroFactura : <$lt:1500> > ).count()
1388
>
```

Consultando una Colección – Criterios de Selección

Busca las facturas cuya condPago sea "CONTADO" ó "30 Ds FF", mostrando sólo los atributos nroFactura y condPago. Explícitamente se debe pedir no mostrar el _id.

Operadores \$

```
db.facturas.find( { condPago: {$in:["CONTADO","30 Ds FF"]} }  
, { _id:0,nroFactura:1,condPago:1})
```

\$gt
\$gte
\$lt
\$lte
\$not
\$or
\$in
\$nin
\$exist
\$regex

```
>  
> db.facturas.find( { condPago: { $in: ["CONTADO","30 Ds FF"] } }, { _id:0,nroFactura:  
:1,condPago:1 } )  
< { "nroFactura" : 1447, "condPago" : "CONTADO" }  
< { "nroFactura" : 1450, "condPago" : "CONTADO" }  
< { "nroFactura" : 1451, "condPago" : "30 Ds FF" }  
< { "nroFactura" : 1453, "condPago" : "CONTADO" }  
< { "nroFactura" : 1454, "condPago" : "CONTADO" }  
< { "nroFactura" : 1455, "condPago" : "30 Ds FF" }  
< { "nroFactura" : 1456, "condPago" : "CONTADO" }  
< { "nroFactura" : 1457, "condPago" : "CONTADO" }  
< { "nroFactura" : 1458, "condPago" : "30 Ds FF" }  
< { "nroFactura" : 1460, "condPago" : "CONTADO" }  
< { "nroFactura" : 1461, "condPago" : "CONTADO" }  
< { "nroFactura" : 1462, "condPago" : "30 Ds FF" }  
< { "nroFactura" : 1463, "condPago" : "CONTADO" }  
< { "nroFactura" : 1464, "condPago" : "CONTADO" }  
< { "nroFactura" : 1465, "condPago" : "30 Ds FF" }  
< { "nroFactura" : 1467, "condPago" : "CONTADO" }  
< { "nroFactura" : 1468, "condPago" : "CONTADO" }  
< { "nroFactura" : 1469, "condPago" : "30 Ds FF" }  
< { "nroFactura" : 1470, "condPago" : "CONTADO" }  
< { "nroFactura" : 1471, "condPago" : "CONTADO" }  
Type "it" for more  
>
```

Consultando una Colección – Criterios de Selección

Busca las facturas de las que el apellido del cliente sea "Manoni" ó condPago sea "60 Ds FF", mostrando sólo los atributos nroFactura y cliente.

Operadores \$

```
db.facturas.find({$or:[{"cliente.apellido":"Manoni"}, {"condPago":"60 Ds FF"}] }  
, {"cliente.apellido":1, condPago:1, _id:0})
```

\$gt
\$gte
\$lt
\$lte
\$not
\$or
\$in
\$nin
\$exist
\$regex

```
> db.facturas.find(<{$or:[{"cliente.apellido":"Manoni"}, {"condPago":"60 Ds FF"}] }, <  
"cliente.apellido":1, condPago:1, _id:0}>  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "CONTADO" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
< "cliente" : < "apellido" : "Manoni" >, "condPago" : "60 Ds FF" >  
Type "it" for more  
>
```

Utilizar el operador \$in en vez del \$or siempre que sea posible, ya que tiene una mejor performance.



Consultando una Colección – Expresiones Regulares

Operador \$exists

Este operador nos permite buscar documentos en los cuáles exista un determinado atributo.

\$exists

Ejemplo - Busca las facturas en los que exista el campo indicador, listando los atributos _id, nombre del cliente e indicador.

```
db.facturas.find( {indicador:{ $exists:true } }, { "cliente.nombre":1, indicador:1} )
```

```
>  
> db.facturas.find(<<indicador:<$exists:true>>,<"cliente.nombre":1,indicador:1>)  
< { "_id" : ObjectId<"53685dbb2baf7b93f61df566">, "cliente" : { "nombre" : "Martin  
" }, "indicador" : "GOOD" }  
>
```