

# Taller de Bases de Datos NoSQL

---

## Objetivo

Crear una base de datos, una tabla y ejecutar operaciones básicas tales como insertar, borrar o actualizar los registros de dicha tabla haciendo uso de MongoDB.

---

## Procedimiento

### 1. Abrir una terminal.

- Para abrir una terminal, siga las instrucciones de los ítems número 1 y 2 de la **Guía de conexión**.

### 2. Abrir MongoDB.

El programa mongoDB ya se encuentra instalado, para abrirlo se escribe en la terminal el comando que aparece a continuación luego del símbolo \$ y presione la tecla ENTER, no importa su ubicación actual.

```
~$ mongo
```

Bloque de código 1

Al presionar la tecla ENTER se abrirá la aplicación en la terminal. La siguiente figura ilustra el inicio de MongoDB, donde el símbolo ">" indica que el programa se encuentra abierto y en espera de comandos. Cada vez que aparezca el símbolo > indica una nueva línea, por tanto una nueva instrucción o comando.

```
cali@cursobigdata: ~
cali@cursobigdata:~$ mongo
MongoDB shell version: 3.0.7
connecting to: test
Server has startup warnings:
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten]
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten] ** WARNING: /sys/kernel/
mm/transparent_hugepage/enabled is 'always'.
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten] **          We suggest set
ting it to 'never'
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten]
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten] ** WARNING: /sys/kernel/
mm/transparent_hugepage/defrag is 'always'.
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten] **          We suggest set
ting it to 'never'
2015-11-27T11:49:30.151-0500 I CONTROL [initandlisten]
>
```

Figura 1

### 3. Ver las bases de datos existentes.

- Para listar las bases de datos existentes digite en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> show dbs
```

Bloque de código 2

### 4. Crear una base de datos.

En MongoDB no existe un comando para crear una base de datos, ésta se crea automáticamente cuando se insertan datos en ella, haciendo uso de una base de datos abstracta con ese nombre. Escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER. Reemplace **DATABASE\_NAME** por el número de su **CÓDIGO**.

```
> use DATABASE_NAME
```

Bloque de código 3

```
cali@cursobigdata: ~
> use DATABASE_NAME
switched to db DATABASE_NAME
>
```

Figura 2

- Para verificar el nombre de la base de datos que está en uso se ingresa en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> db
```

Bloque de código 4

Luego de ejecutar esta instrucción, debe aparecer su **CÓDIGO**, ya que es el nombre de la base de datos que ésta en uso.

## 5. Crear una colección en la base de datos que ésta en uso.

- Para crear una colección dentro de la base de datos abstracta que ésta en uso escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER. Reemplace **COLLECTION\_NAME** por el nombre de su preferencia y conserve las comillas existentes ( " " ) .

```
> db.createCollection("COLLECTION_NAME")
```

Bloque de código 5

- Una vez creada la colección dentro de la base de datos que ésta en uso, escriba en la terminal el comando que aparece en el **Bloque de código 2** luego del símbolo > y presione la tecla ENTER para listar las bases de datos existentes. Compruebe la existencia de la base de datos ya creada con su **CÓDIGO**.

## 6. Ver las colecciones existentes.

- Verifique la existencia de la colección creada, listando las colecciones existentes. Escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> show collections
```

Bloque de código 6

## 7. Renombrar la colección creada.

- Para renombrar la colección creada, escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER. Reemplace **COLLECTION\_NAME** por el nombre que asignó previamente a la

colección y **NEW\_NAME\_COLLECTION** por el nuevo nombre que desea asignar, conserve las comillas existentes (“ ”) .

```
> db.COLLECTION_NAME.renameCollection("NEW_NAME_COLLECTION")
```

Bloque de código 7

- Verifique el cambio de nombre de la colección, listando las colecciones existentes escribiendo en la terminal el comando que aparece en el **Bloque de código 6** luego del símbolo > y presione la tecla ENTER.

## 8. Eliminar la colección renombrada.

- Para eliminar la colección que creó y posteriormente renombró, escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER. Reemplace **NAME\_COLLECTION** por el nombre que asignó a la colección a la hora de renombrarla.

```
> db.NAME_COLLECTION.drop()
```

Bloque de código 8

- Para verificar que la colección ha sido eliminada, liste las colecciones existentes escribiendo en la terminal el comando que aparece en el **Bloque de código 6** luego del símbolo > y presione la tecla ENTER.

## 9. Importar un dataset.

- Para importar un dataset a su base de datos desde la terminal abierta debe cerrarse MongoDB, escribiendo en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> exit
```

Bloque de código 9

- Para importar el dataset debe tenerse en cuenta la ubicación del archivo y su correspondiente ruta. El dataset ubicado en **/home/cali/Files/dataset.json** será importado a la base de datos previamente creada con su **CÓDIGO** a una colección llamada **restaurants**. Para esto, se ingresa en la terminal el comando que aparece a continuación luego del símbolo \$ y presione la tecla ENTER. Reemplace **DATABASE\_NAME** por el nombre de su base de datos (su **CÓDIGO**).

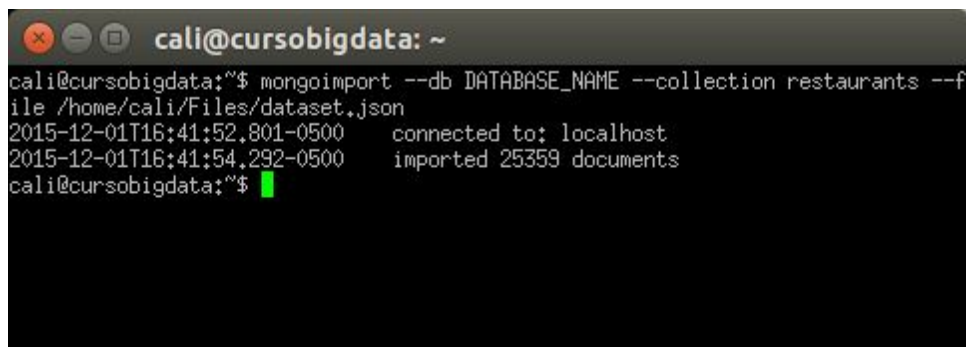
```
~$ mongoimport --db DATABASE_NAME --collection restaurants  
--file /home/cali/Files/dataset.json
```

Bloque de código 10

Ya que no existe una colección llamada **restaurants** en la base de datos, al importar el dataset solo estarán presentes los datos correspondientes a éste. De lo contrario, si existiera una colección previamente creada con el nombre **restaurants**, ésta puede eliminarse junto con su contenido haciendo uso del comando **--drop** de la siguiente manera.

```
~$ mongoimport --db DATABASE_NAME --collection restaurants  
--drop --file /home/cali/Files/dataset.json
```

Bloque de código 11

A terminal window titled 'cali@cursobigdata: ~' shows the execution of the 'mongoimport' command. The command is: 'mongoimport --db DATABASE\_NAME --collection restaurants --file /home/cali/Files/dataset.json'. The output shows two lines of status information: '2015-12-01T16:41:52.801-0500 connected to: localhost' and '2015-12-01T16:41:54.292-0500 imported 25359 documents'. The prompt 'cali@cursobigdata:~\$' is followed by a green cursor.

```
cali@cursobigdata: ~  
cali@cursobigdata:~$ mongoimport --db DATABASE_NAME --collection restaurants --f  
ile /home/cali/Files/dataset.json  
2015-12-01T16:41:52.801-0500    connected to: localhost  
2015-12-01T16:41:54.292-0500    imported 25359 documents  
cali@cursobigdata:~$
```

Figura 3

Como se puede observar, han sido importados 25359 documentos a la base de datos **DATABASE\_NAME** y a la colección **restaurants**.

- Una vez importado el dataset debe ingresar en la terminal el comando que aparece a continuación luego del símbolo **\$** y presionar la tecla ENTER para abrir nuevamente MongoDB.

```
~$ mongo
```

Bloque de código 12

Posteriormente, use la base de datos creada con su **CÓDIGO** escribiendo en la terminal el comando que aparece a continuación luego del símbolo **>** y presione la tecla ENTER. Reemplace **DATABASE\_NAME** por su **CÓDIGO**.

```
> use DATABASE_NAME
```

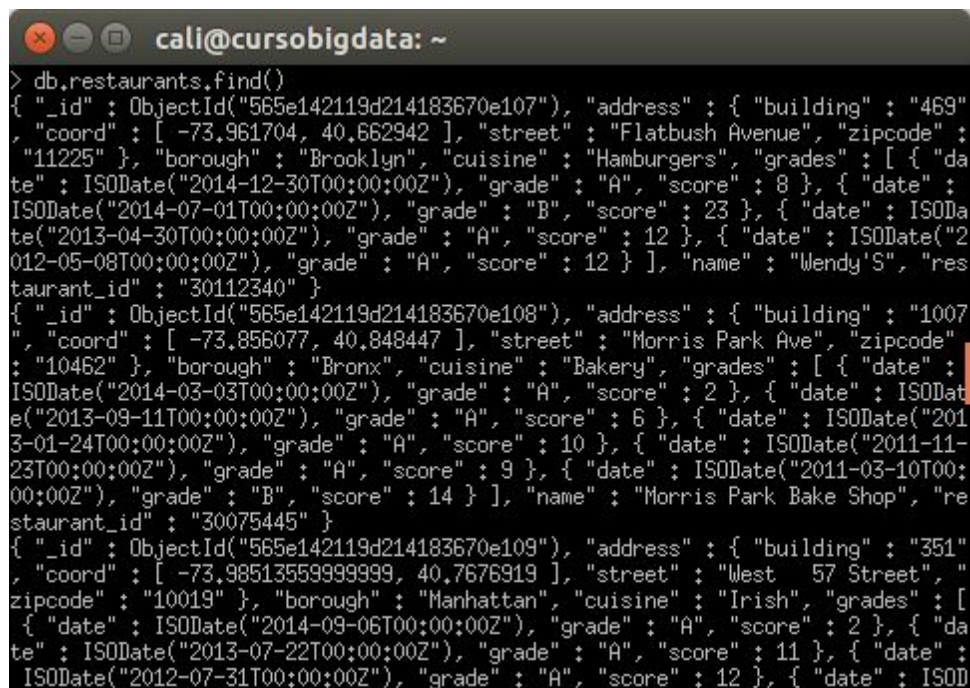
Bloque de código 13

## 10. Visualizar y contar los documentos que contiene la colección **restaurants**.

- Para visualizar los documentos que tiene la colección **restaurants** contenida en su base de datos, se escribe en la terminal el comando que aparece a continuación luego del símbolo **>** y presione la tecla ENTER.

```
> db.restaurants.find()
```

Bloque de código 14



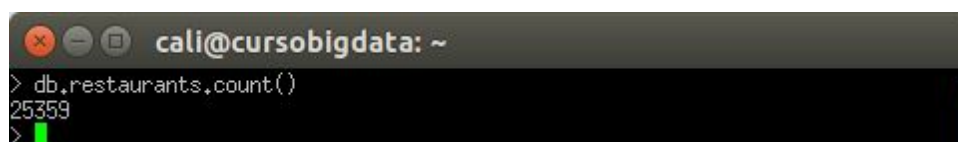
```
cali@cursobigdata: ~  
> db.restaurants.find()  
{ "_id" : ObjectId("565e142119d214183670e107"), "address" : { "building" : "469"  
  , "coord" : [ -73.961704, 40.662942 ], "street" : "Flatbush Avenue", "zipcode" :  
  "11225" }, "borough" : "Brooklyn", "cuisine" : "Hamburgers", "grades" : [ { "da  
te" : ISODate("2014-12-30T00:00:00Z"), "grade" : "A", "score" : 8 }, { "date" :  
ISODate("2014-07-01T00:00:00Z"), "grade" : "B", "score" : 23 }, { "date" : ISODate(  
"2013-04-30T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2  
012-05-08T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "Wendy'S", "res  
taurant_id" : "30112340" }  
{ "_id" : ObjectId("565e142119d214183670e108"), "address" : { "building" : "1007"  
  , "coord" : [ -73.856077, 40.848447 ], "street" : "Morris Park Ave", "zipcode"  
  : "10462" }, "borough" : "Bronx", "cuisine" : "Bakery", "grades" : [ { "date" :  
ISODate("2014-03-03T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate(  
"2013-09-11T00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("201  
3-01-24T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-11-  
23T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2011-03-10T00:  
00:00Z"), "grade" : "B", "score" : 14 } ], "name" : "Morris Park Bake Shop", "re  
staurant_id" : "30075445" }  
{ "_id" : ObjectId("565e142119d214183670e109"), "address" : { "building" : "351"  
  , "coord" : [ -73.98513559999999, 40.7676919 ], "street" : "West 57 Street", "  
zipcode" : "10019" }, "borough" : "Manhattan", "cuisine" : "Irish", "grades" : [  
  { "date" : ISODate("2014-09-06T00:00:00Z"), "grade" : "A", "score" : 2 }, { "da  
te" : ISODate("2013-07-22T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" :  
ISODate("2012-07-31T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-07-31T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-07-31T00:00:00Z"), "grade" : "A", "score" : 12 } ], "name" : "West 57 Irish Pub", "restaurant_id" : "30075446" }
```

Figura 4

- Para contar los documentos existentes que tiene la colección **restaurants** contenida en su base de datos, se escribe en la terminal el comando que aparece a continuación luego del símbolo **>** y presione la tecla ENTER. El número de documentos deberá ser **25359**.

```
> db.restaurants.count()
```

Bloque de código 15



```
cali@cursobigdata: ~  
> db.restaurants.count()  
25359  
>
```

Figura 5

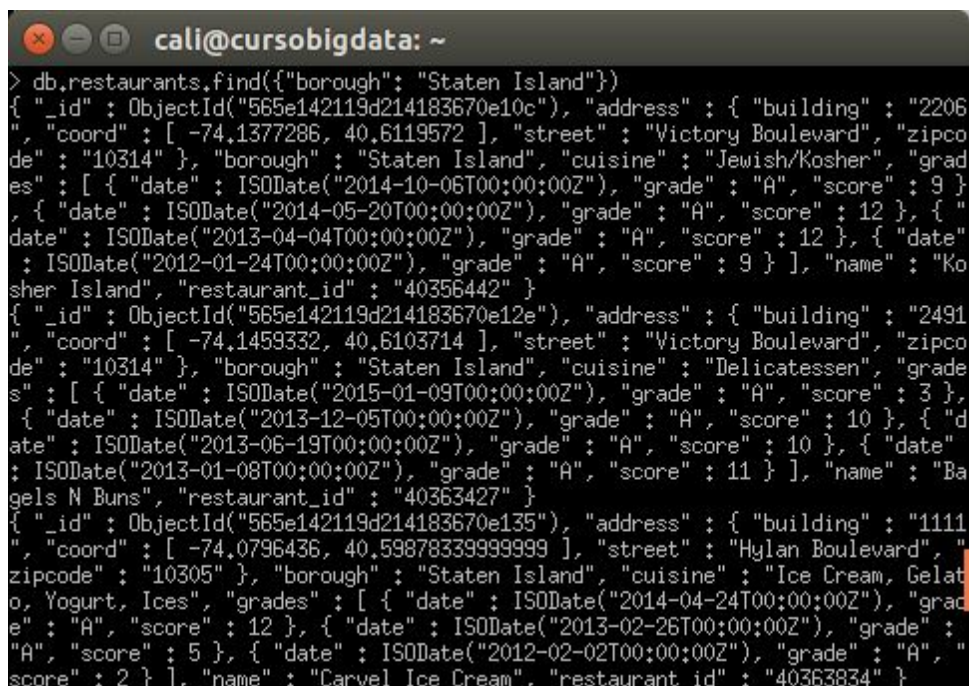
## 11. Realizar búsquedas en la colección restaurants.

- Para realizar una búsqueda en la colección se filtran los campos necesarios de los existentes en cada uno de los documentos, entre ellos el identificador único de un documento dentro de la base de datos denominado `_id`.

- Buscar los documentos que tienen en el campo **"borough"** el valor **"Staten Island"**. Se escribe en la terminal el comando que aparece a continuación luego del símbolo `>` y presione la tecla ENTER. Parte del resultado de dicha query o consulta se muestra en la **Figura 6**.

```
> db.restaurants.find({"borough": "Staten Island"})
```

Bloque de código 16



```
cali@cursobigdata: ~
> db.restaurants.find({"borough": "Staten Island"})
{ "_id" : ObjectId("565e142119d214183670e10c"), "address" : { "building" : "2206", "coord" : [ -74.1377286, 40.6119572 ], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuisine" : "Jewish/Kosher", "grades" : [ { "date" : ISODate("2014-10-06T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2014-05-20T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-04-04T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-01-24T00:00:00Z"), "grade" : "A", "score" : 9 } ], "name" : "Kosher Island", "restaurant_id" : "40356442" }
{ "_id" : ObjectId("565e142119d214183670e12e"), "address" : { "building" : "2491", "coord" : [ -74.1459332, 40.6103714 ], "street" : "Victory Boulevard", "zipcode" : "10314" }, "borough" : "Staten Island", "cuisine" : "Delicatessen", "grades" : [ { "date" : ISODate("2015-01-09T00:00:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISODate("2013-12-05T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-06-19T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2013-01-08T00:00:00Z"), "grade" : "A", "score" : 11 } ], "name" : "Bagels N Buns", "restaurant_id" : "40363427" }
{ "_id" : ObjectId("565e142119d214183670e135"), "address" : { "building" : "1111", "coord" : [ -74.0796436, 40.59878339999999 ], "street" : "Hylan Boulevard", "zipcode" : "10305" }, "borough" : "Staten Island", "cuisine" : "Ice Cream, Gelato, Yogurt, Ices", "grades" : [ { "date" : ISODate("2014-04-24T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2013-02-26T00:00:00Z"), "grade" : "A", "score" : 5 }, { "date" : ISODate("2012-02-02T00:00:00Z"), "grade" : "A", "score" : 2 } ], "name" : "Carvel Ice Cream", "restaurant_id" : "40363834" }
```

Figura 6

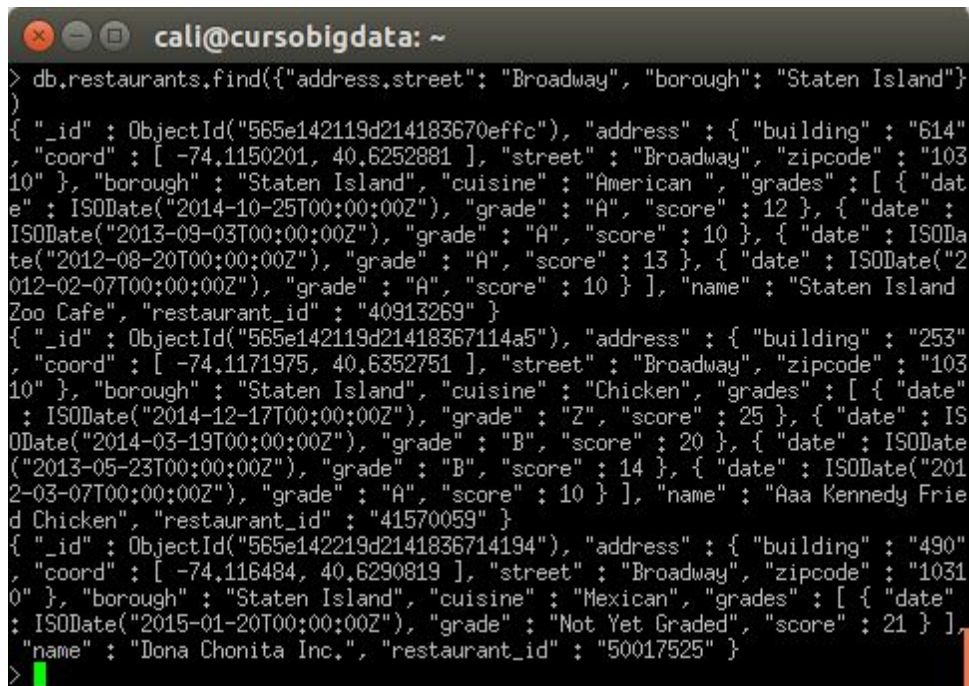
- Buscar los documentos que tienen en el campo **"borough"** el valor **"Staten Island"** y en el campo **"address.street"** el valor **"Broadway"**. Se escribe en la terminal el comando que aparece a continuación luego del símbolo `>` y presione la tecla ENTER.

```
> db.restaurants.find({"address.street": "Broadway", "borough": "Staten Island"})
```

Bloque de código 17



El resultado de dicha query o consulta se muestra en la **Figura 7**, donde podrá identificar tres (3) documentos que cumplen con dichas condiciones.



```
cali@cursobigdata: ~
> db.restaurants.find({"address.street": "Broadway", "borough": "Staten Island"})
{ "_id" : ObjectId("565e142119d214183670effc"), "address" : { "building" : "614",
"coord" : [ -74.1150201, 40.6252881 ], "street" : "Broadway", "zipcode" : "103
10" }, "borough" : "Staten Island", "cuisine" : "American ", "grades" : [ { "dat
e" : ISODate("2014-10-25T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" :
ISODate("2013-09-03T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODa
te("2012-08-20T00:00:00Z"), "grade" : "A", "score" : 13 }, { "date" : ISODate("2
012-02-07T00:00:00Z"), "grade" : "A", "score" : 10 } ], "name" : "Staten Island
Zoo Cafe", "restaurant_id" : "40913269" }
{ "_id" : ObjectId("565e142119d21418367114a5"), "address" : { "building" : "253"
, "coord" : [ -74.1171975, 40.6352751 ], "street" : "Broadway", "zipcode" : "103
10" }, "borough" : "Staten Island", "cuisine" : "Chicken", "grades" : [ { "date"
: ISODate("2014-12-17T00:00:00Z"), "grade" : "Z", "score" : 25 }, { "date" : IS
ODate("2014-03-19T00:00:00Z"), "grade" : "B", "score" : 20 }, { "date" : ISODate
("2013-05-23T00:00:00Z"), "grade" : "B", "score" : 14 }, { "date" : ISODate("201
2-03-07T00:00:00Z"), "grade" : "A", "score" : 10 } ], "name" : "Aaa Kennedy Frie
d Chicken", "restaurant_id" : "41570059" }
{ "_id" : ObjectId("565e142219d2141836714194"), "address" : { "building" : "490"
, "coord" : [ -74.116484, 40.6290819 ], "street" : "Broadway", "zipcode" : "1031
0" }, "borough" : "Staten Island", "cuisine" : "Mexican", "grades" : [ { "date"
: ISODate("2015-01-20T00:00:00Z"), "grade" : "Not Yet Graded", "score" : 21 } ],
"name" : "Dona Chonita Inc.", "restaurant_id" : "50017525" }
>
```

Figura 7

## 12. Eliminar el primer documento listado en el ítem anterior.

- Eliminar el primer documento que cumple con las condiciones previstas en el ítem anterior en una operación que puede hacerse de varias formas:

- Para eliminar el **primer** documento listado que cumple con las condiciones previstas anteriormente se puede ingresar en la terminal cualquiera de las dos opciones siguientes y presione la tecla ENTER.

```
> db.restaurants.remove({"address.street" : "Broadway",
"borough" : "Staten Island"}, 1)
```

Bloque de código 18

```
> db.restaurants.remove({"address.street" : "Broadway",
"borough" : "Staten Island"}, {justOne: true})
```

Bloque de código 19



Luego de ejecutar una de las instrucciones previas para eliminar el primer documento listado que cumple con las condiciones señaladas deberá visualizar lo que aparece en la figura que se muestra a continuación.



```
cali@cursobigdata: ~  
> db.restaurants.remove({"address.street": "Broadway", "borough": "Staten Island"}, 1)  
WriteResult({ "nRemoved" : 1 })  
>
```

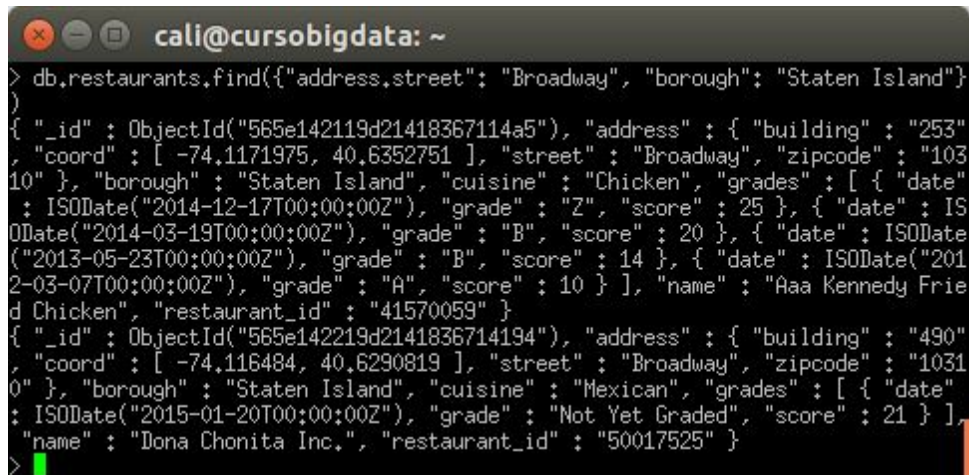
Figura 8

- Para eliminar el primer documento listado valiéndose un campo y valor único para cada documento presente en el dataset, puede emplearse por ejemplo el campo **restaurant\_id**. Escriba en la terminal el comando que aparece a continuación luego del símbolo **>** y presione la tecla ENTER.

```
> db.restaurants.remove({"restaurant_id" : "40913269"})
```

Bloque de código 20

Cualquiera que sea la opción elegida, al visualizar los documentos que tienen en el campo "borough" "Staten Island" y en el campo "address.street" "Broadway" escribiendo en la terminal la instrucción presente en el **Bloque de código 17**, aparecerán listados únicamente dos documentos ya que fue eliminado uno, tal como se muestra en la **Figura 9**.



```
cali@cursobigdata: ~  
> db.restaurants.find({"address.street": "Broadway", "borough": "Staten Island"})  
{ "_id" : ObjectId("565e142119d21418367114a5"), "address" : { "building" : "253",  
  "coord" : [ -74.1171975, 40.6352751 ], "street" : "Broadway", "zipcode" : "10310" },  
  "borough" : "Staten Island", "cuisine" : "Chicken", "grades" : [ { "date" : ISODate("2014-12-17T00:00:00Z"), "grade" : "Z", "score" : 25 }, { "date" : ISODate("2014-03-19T00:00:00Z"), "grade" : "B", "score" : 20 }, { "date" : ISODate("2013-05-23T00:00:00Z"), "grade" : "B", "score" : 14 }, { "date" : ISODate("2012-03-07T00:00:00Z"), "grade" : "A", "score" : 10 } ], "name" : "Aaa Kennedy Fried Chicken", "restaurant_id" : "41570059" }  
{ "_id" : ObjectId("565e142219d2141836714194"), "address" : { "building" : "490",  
  "coord" : [ -74.116484, 40.6290819 ], "street" : "Broadway", "zipcode" : "10310" },  
  "borough" : "Staten Island", "cuisine" : "Mexican", "grades" : [ { "date" : ISODate("2015-01-20T00:00:00Z"), "grade" : "Not Yet Graded", "score" : 21 } ],  
  "name" : "Dona Chonita Inc.", "restaurant_id" : "50017525" }  
>
```

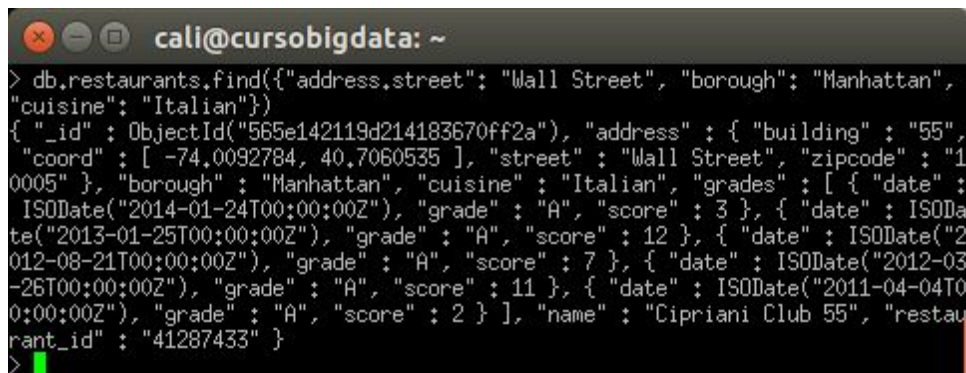
Figura 9

### 13. Actualizar un campo de un documento de la colección restaurants.

- Primero se debe definir el documento a actualizar. Al realizar una consulta de los restaurantes ubicados en Wall Street - Manhattan que preparen comida italiana se encontró un único documento, que en este caso será modificado. Escribe en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> db.restaurants.find({"address.street": "Wall Street",  
"borough": "Manhattan", "cuisine": "Italian"})
```

Bloque de código 21



```
cali@cursobigdata: ~  
> db.restaurants.find({"address.street": "Wall Street", "borough": "Manhattan",  
"cuisine": "Italian"})  
{ "_id" : ObjectId("565e142119d214183670ff2a"), "address" : { "building" : "55",  
"coord" : [ -74.0092784, 40.7060535 ], "street" : "Wall Street", "zipcode" : "1  
0005" }, "borough" : "Manhattan", "cuisine" : "Italian", "grades" : [ { "date" :  
ISODate("2014-01-24T00:00:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISODate("2013-01-25T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("2012-08-21T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-03-26T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-04-04T00:00:00Z"), "grade" : "A", "score" : 2 } ], "name" : "Cipriani Club 55", "restaurant_id" : "41287433" }  
>
```

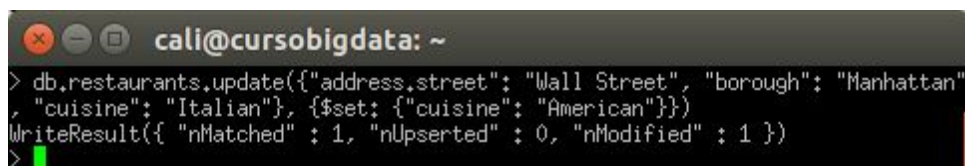
Figura 10

Al documento resultante (**Figura 10**) identificado por el **restaurant\_id** numero **41287433**, el campo que se actualizará será el tipo de cocina, de italiana a americana. Escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> db.restaurants.update({"address.street": "Wall Street",  
"borough": "Manhattan", "cuisine": "Italian"}, {$set:  
{"cuisine": "American"}})
```

Bloque de código 22

Luego de presionar la tecla ENTER deberá visualizar lo que aparece en la figura que se muestra a continuación.



```
cali@cursobigdata: ~  
> db.restaurants.update({"address.street": "Wall Street", "borough": "Manhattan",  
"cuisine": "Italian"}, {$set: {"cuisine": "American"}})  
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })  
>
```

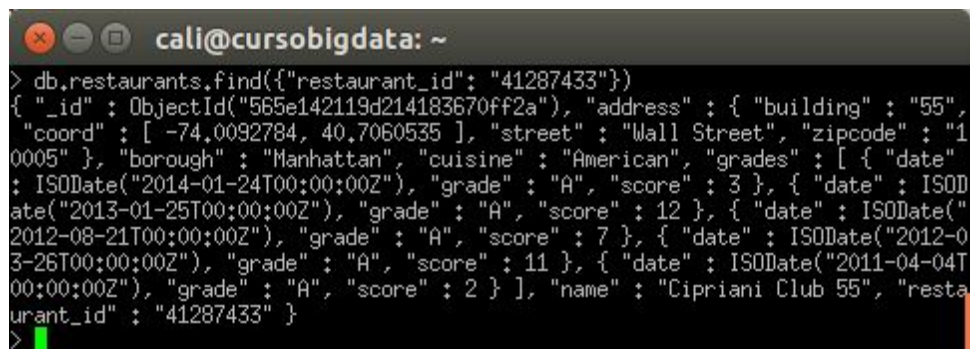
Figura 11

- Para verificar el cambio realizado se efectúa la búsqueda del documento con ayuda del **restaurant\_id**, escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> db.restaurants.find({"restaurant_id" : "41287433"})
```

Bloque de código 23

Como se puede observar en la figura que se muestra a continuación, el campo **cuisine** que inicialmente tenía como valor **Italian** ahora tiene **American**, mientras que todos los campos restantes del documento se conservan idénticos.



```
cali@cursobigdata: ~  
> db.restaurants.find({"restaurant_id": "41287433"})  
{ "_id" : ObjectId("565e142119d214183670ff2a"), "address" : { "building" : "55",  
  "coord" : [ -74.0092784, 40.7060535 ], "street" : "Wall Street", "zipcode" : "1  
0005" }, "borough" : "Manhattan", "cuisine" : "American", "grades" : [ { "date"  
: ISODate("2014-01-24T00:00:00Z"), "grade" : "A", "score" : 3 }, { "date" : ISO  
Date("2013-01-25T00:00:00Z"), "grade" : "A", "score" : 12 }, { "date" : ISODate("  
2012-08-21T00:00:00Z"), "grade" : "A", "score" : 7 }, { "date" : ISODate("2012-0  
3-26T00:00:00Z"), "grade" : "A", "score" : 11 }, { "date" : ISODate("2011-04-04T  
00:00:00Z"), "grade" : "A", "score" : 2 } ], "name" : "Cipriani Club 55", "resta  
urant_id" : "41287433" }  
>
```

Figura 12

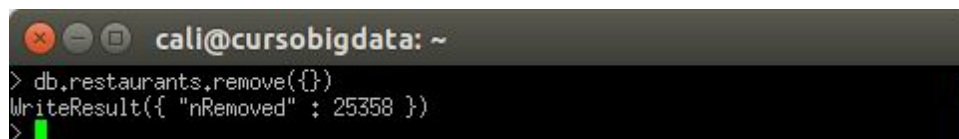
## 14. Eliminar todos los documentos de una colección sin eliminarla.

- Para eliminar todos los documentos de la colección **restaurants** sin eliminarla, escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> db.restaurants.remove({})
```

Bloque de código 24

Luego de presionar la tecla ENTER deberá visualizar lo que aparece en la figura que se muestra a continuación. Se puede observar que han sido eliminados **25358** documentos de **25359** que inicialmente fueron importados ya que se eliminó uno de ellos previamente.



```
cali@cursobigdata: ~  
> db.restaurants.remove({})  
WriteResult({ "nRemoved" : 25358 })  
>
```

Figura 13

- Para verificar que la colección **restaurants** aún existe, liste las colecciones existentes escribiendo en la terminal el comando que aparece luego del símbolo > y presione la tecla ENTER.

```
> show collections
```

Bloque de código 25

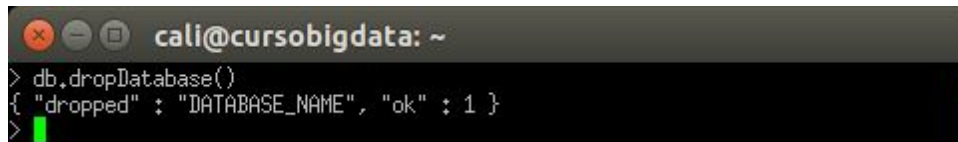
## 15. Eliminar la base de datos.

- Para eliminar la base de datos junto con su contenido, escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> db.dropDatabase()
```

Bloque de código 26

Luego de presionar la tecla ENTER deberá visualizar lo que aparece en la figura que se muestra a continuación.



```
cali@kursobigdata: ~  
> db.dropDatabase()  
{ "dropped" : "DATABASE_NAME", "ok" : 1 }  
>
```

Figura 14

- Verificar que la base de datos no existe listando las bases de datos existentes. Escriba en la terminal el comando que aparece a continuación luego del símbolo > y presione la tecla ENTER.

```
> show dbs
```

Bloque de código 27

## 16. Cierre la terminal y desconéctese del servidor.

- Para desconectarse del servidor, siga las instrucciones de los ítems número 5 y 6 de la **Guía de conexión**.

---

**Fin de la Guía**