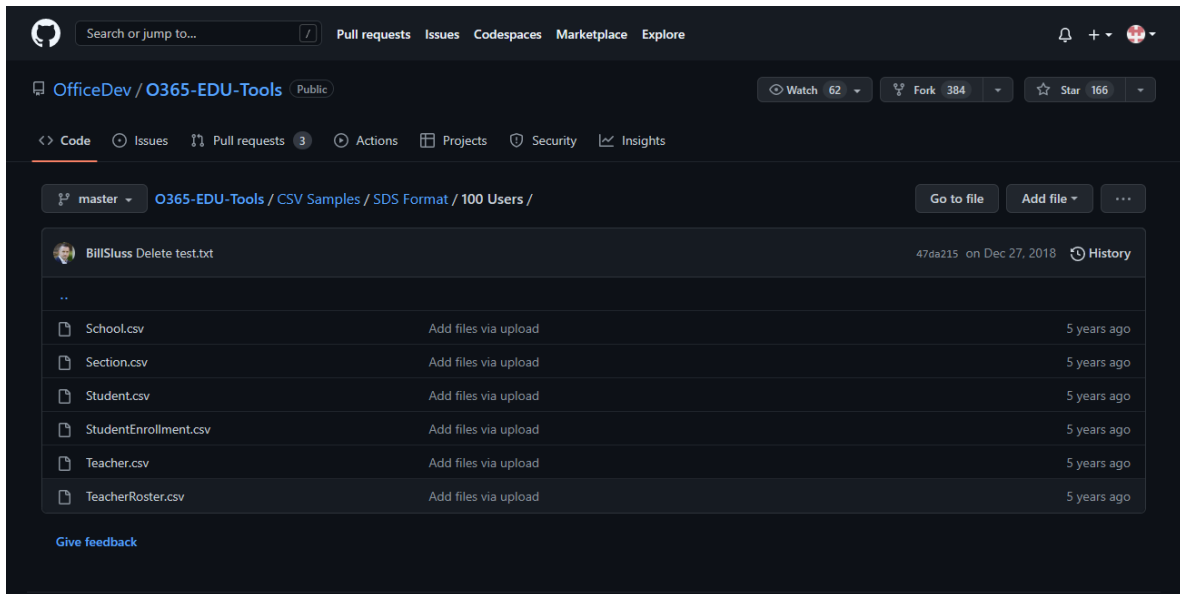


A continuación, se mostrará el paso a paso del trabajo con MongoDB.

Se entró a GitHub y desde ahí se descargo un archivo csv de un repositorio común llamado OfficeDev, (en este caso se descargo el csv de School).



Luego de tener el archivo csv se entró a un cambiador de extensión de csv a json

### Step 1: Select your input

Enter Data **Choose File** Enter URL

Enter or paste CSV here New - duplicate column names are converted into an Array

Clear Input **Example 1** Example 2

## Step 1: Select your input

Enter Data Choose File Enter URL

Choose File Elegir archivo Student.csv Encoding -Default-

Clear Input Example 1 Example 2

Input Records- Header: true Header Fields: 14  
Data Separator: , Fields: 14 Records: 22

## Step 5: Generate output

Choose Conversion Type:

CSV To JSON CSV To Keyed JSON CSV To JSON Array CSV To JSON Column Array

Result Data:

```
{
  "SIS ID": 13001,
  "School SIS ID": 10001,
  "First Name": "Ora",
  "Last Name": "Klein",
  "Username": "OKlein",
  "Password": "P@ssword",
  "State ID": "WA",
  "Secondary Email": "",
  "Student Number": 13001,
  "Middle Name": "Christopher",
  "Grade": 9,
```

Luego de la conversión de archivo, se validó en Json Formatter que estaba correcto el archivo

JSON formatter

JSON BEAUTIFIER JSON PARSER XML FORMATTER JSBEAUTIFIER SAVE RECENT LINKS LOGIN

Upload Data

Validate

2 Tab Space

Format / Beautify

Advertisements

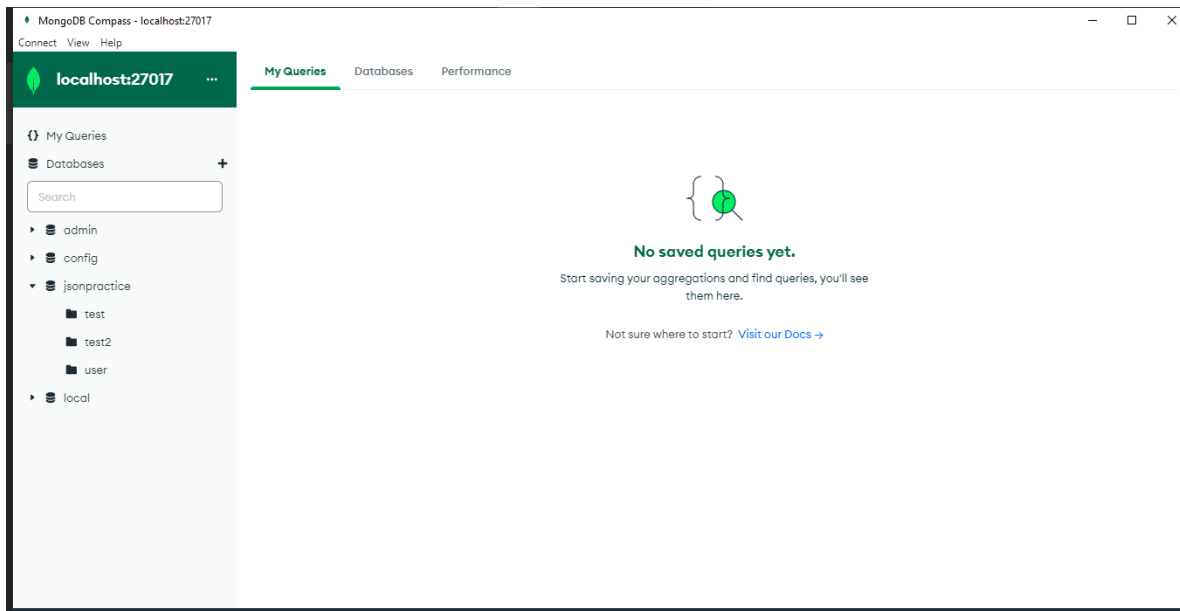
Minify / Compact

```
330 "Secondary Email": "",
331 "Student Number": 13026,
332 "Middle Name": "Chris",
333 "Grade": 11,
334 "Status": "Active",
335 "Birthdate": "8/12/1999",
336 "Graduation Year": 2018
337 },
338 {
339   "SIS ID": 13022,
340   "School SIS ID": 10001,
341   "First Name": "Williams",
342   "Last Name": "Bisson",
343   "Username": "Wbisson",
344   "Password": "P@ssword",
345   "State ID": "WA",
346   "Secondary Email": "",
347   "Student Number": 13027,
348   "Middle Name": "Roxanne",
349   "Grade": 9,
350   "Status": "Active",
351   "Birthdate": "3/6/2001",
352   "Graduation Year": 2020
353 }
354 ]
```

```
1 {
2   "SIS ID": 13001,
3   "School SIS ID": 10001,
4   "First Name": "Ora",
5   "Last Name": "Klein",
6   "Username": "OKlein",
7   "Password": "P@ssword",
8   "State ID": "WA",
9   "Secondary Email": "",
10  "Student Number": 13001,
11  "Middle Name": "Christopher",
12  "Grade": 9,
13  "Status": "Active",
14  "Birthdate": "4/2/2000",
15  "Graduation Year": 2019
16 },
17 {
18   "SIS ID": 13002,
19   "School SIS ID": 10001,
20   "First Name": "Beulah",
21   "Last Name": "McMillan",
22   "Username": "BMcMillan",
23   "Password": "P@ssword",
24   "State ID": "WA"
```

A partir de ahí ya se puede iniciar a trabajar.

Se abre mongoDb Compass y se conecta al LocalHost



Se abre el primer cmd que ubicará la ruta de la base tal que así:

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.19045.2604]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\02>cd ..
C:\Users>cd ..
C:\>cd MongoDB
C:\MongoDB>cd Server
C:\MongoDB\Server>cd 5.0
C:\MongoDB\Server\5.0>cd bin
C:\MongoDB\Server\5.0\bin>
```

Luego escribiríamos el comando que haría el llamado a la base.

```
C:\MongoDB\Server\5.0\bin>mongod.exe
```

A continuación, se debería ver así y no parar su ejecución

```

C:\Windows\system32\cmd.exe - mongod.exe --dbpath c:\test\data\db
{"version":0,"maxWireVersion":13,"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":13},
{"outgoing":{"minWireVersion":0,"maxWireVersion":13},"isInternalClient":true}}}
{"t":{"$date":"2023-03-06T20:32:15.859-05:00"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"
thread1","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabled
Protocols 'none'"}
{"t":{"$date":"2023-03-06T20:32:15.861-05:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"
thread1","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2023-03-06T20:32:15.861-05:00"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"
thread1","msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2023-03-06T20:32:15.863-05:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"
thread1","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2023-03-06T20:32:15.863-05:00"},"s":"I", "c":"REPL", "id":5123008, "ctx":"
thread1","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigration
DonorService","ns":"config.tenantMigrationDonors"}}
{"t":{"$date":"2023-03-06T20:32:15.863-05:00"},"s":"I", "c":"REPL", "id":5123008, "ctx":"
thread1","msg":"Successfully registered PrimaryOnlyService","attr":{"service":"TenantMigration
RecipientService","ns":"config.tenantMigrationRecipients"}}
{"t":{"$date":"2023-03-06T20:32:15.863-05:00"},"s":"I", "c":"CONTROL", "id":5945603, "ctx":"
thread1","msg":"Multi threading initialized"}
{"t":{"$date":"2023-03-06T20:32:15.866-05:00"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"
initandlisten","msg":"MongoDB starting","attr":{"pid":10280,"port":27017,"dbPath":"C:/data/db/
","architecture":"64-bit","host":"DESKTOP-DCA76SJ"}}
{"t":{"$date":"2023-03-06T20:32:15.866-05:00"},"s":"I", "c":"CONTROL", "id":23398, "ctx":"
initandlisten","msg":"Target operating system minimum version","attr":{"targetMinOS":"Windows
7/Windows Server 2008 R2"}}
{"t":{"$date":"2023-03-06T20:32:15.866-05:00"},"s":"I", "c":"CONTROL", "id":23403, "ctx":"
initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"5.0.15","gitVersion":"935639
beed3d0c19c2551c93854b831107c0b118","modules":[],"allocator":"tcmalloc","environment":{"distmo
d":"windows","distarch":"x86_64","target_arch":"x86_64"}}}}}
{"t":{"$date":"2023-03-06T20:32:15.867-05:00"},"s":"I", "c":"CONTROL", "id":51765, "ctx":"

```

Luego se ejecuta un nuevo cmd donde correremos la base de datos y empezaremos a interactuar con ella.

```

Microsoft Windows [Versión 10.0.19045.2604]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\02>cd ..

C:\Users>cd ..

C:\>cd MongoDB

C:\MongoDB>cd Server

C:\MongoDB\Server>cd 5.0

C:\MongoDB\Server\5.0>cd bin

C:\MongoDB\Server\5.0\bin>mongo.exe

```

```

---
The server generated these startup warnings when booting:
  2023-03-02T11:23:48.252-05:00: Access control is not enabled for the database. Read and write access to data and
configuration is unrestricted
---
  Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

  The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

  To enable free monitoring, run the following command: db.enableFreeMonitoring()
  To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
>

```

Se empiezan a usar comandos básicos donde crearemos Base de datos y colección

```
> use jsonpractice
```

Crea la base de datos

```

> show dbs
admin                0.000GB
config               0.000GB
jsonpractice         0.000GB
local                0.000GB
> _

```

Muestra las bases de datos actuales.

```

> db.createCollection("locapso");
{ "ok" : 1 }
>

```

Crear una colección

```

> show collections
locapso
> _

```

Muestra las colecciones existentes.

En el caso de la base que estamos armando usamos estamos collections

```

> show collections
test
test2
user

```

Lo siguiente seria hacer la conexión de el json convertido a la base de datos ya que el ya nos ofrece datos.

Tenemos que abrir un nuevo cmd que se ubique en la ruta de la base de datos y desde ahí hacer el comando de llamado

```
mongoimport --jsonArray --db test --collection docs --file example2.json
```

Este es un ejemplo tomado de StackOverflow en nuestro caso quedaría de esta manera.

Es de aclarar que antes:

- Se debe descargar mongoimport y colocar en la ruta de bin
- Se debe crear una base de datos
- Se debe crear una colección
- El (--jsonArray) se usa para evitar errores y se ajuste a un Array con datos.

Se empiezan a validar las primeras consultas

En este caso muestra la primera consulta

```
> db.test2.findOne()
{
  "_id" : ObjectId("64069c6f11cd57574309cff4"),
  "SIS ID" : 13001,
  "School SIS ID" : 10001,
  "First Name" : "Ora",
  "Last Name" : "Klein",
  "Username" : "OKlein",
  "Password" : "P@ssword",
  "State ID" : "WA",
  "Secondary Email" : "",
  "Student Number" : 13001,
  "Middle Name" : "Christopher",
  "Grade" : 9,
  "Status" : "Active",
  "Birthdate" : "4/2/2000",
  "Graduation Year" : 2019
}
```

Mostrar todos los datos que se tiene en el .json

```

    "Graduation Year" : 2019 }
}
> db.test2.find()
{ "_id" : ObjectId("64069c6f11cd57574309cff4"), "SIS ID" : 13001, "School SIS ID" :
10001, "First Name" : "Ora", "Last Name" : "Klein", "Username" : "OKlein", "Passwo
rd" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 130
01, "Middle Name" : "Christopher", "Grade" : 9, "Status" : "Active", "Birthdate" :
"4/2/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cff5"), "SIS ID" : 13006, "School SIS ID" :
10001, "First Name" : "Sherry", "Last Name" : "Santana", "Username" : "SSantana",
"Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number
" : 13006, "Middle Name" : "Thomas", "Grade" : 10, "Status" : "Active", "Birthdate"
: "8/22/1999", "Graduation Year" : 2018 }
{ "_id" : ObjectId("64069c6f11cd57574309cff6"), "SIS ID" : 13005, "School SIS ID" :
10001, "First Name" : "Erna", "Last Name" : "Parker", "Username" : "EParker", "Pas
sword" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" :
13005, "Middle Name" : "Troy", "Grade" : 11, "Status" : "Active", "Birthdate" : "5/
17/1998", "Graduation Year" : 2017 }
{ "_id" : ObjectId("64069c6f11cd57574309cff7"), "SIS ID" : 13003, "School SIS ID" :
10001, "First Name" : "Florence", "Last Name" : "Stark", "Username" : "FStark", "P
assword" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number"
: 13003, "Middle Name" : "Brian", "Grade" : 12, "Status" : "Active", "Birthdate" :
"12/19/1997", "Graduation Year" : 2016 }
{ "_id" : ObjectId("64069c6f11cd57574309cff8"), "SIS ID" : 13017, "School SIS ID" :
10001, "First Name" : "Maribel", "Last Name" : "Parsons", "Username" : "Mparsons",
"Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Numbe
r" : 13022, "Middle Name" : "Delmar", "Grade" : 10, "Status" : "Active", "Birthdate

```

Muestra los usuarios que se encuentran en grado noveno y decimo

```
> db.test2.find({"Grade":{"$in": [9,10]}));
{ "_id" : ObjectId("64069c6f11cd57574309cff4"), "SIS ID" : 13001, "School SIS ID" : 10001, "First Name" : "Ora", "Last Name" : "Klein", "Username" : "OKlein", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13001, "Middle Name" : "Christopher", "Grade" : 9, "Status" : "Active", "Birthdate" : "4/2/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cff5"), "SIS ID" : 13006, "School SIS ID" : 10001, "First Name" : "Sherry", "Last Name" : "Santana", "Username" : "SSantana", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13006, "Middle Name" : "Thomas", "Grade" : 10, "Status" : "Active", "Birthdate" : "8/22/1999", "Graduation Year" : 2018 }
{ "_id" : ObjectId("64069c6f11cd57574309cff8"), "SIS ID" : 13017, "School SIS ID" : 10001, "First Name" : "Maribel", "Last Name" : "Parsons", "Username" : "Mparsons", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13022, "Middle Name" : "Delmar", "Grade" : 10, "Status" : "Active", "Birthdate" : "3/10/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cffa"), "SIS ID" : 13019, "School SIS ID" : 10001, "First Name" : "Wilfred", "Last Name" : "Bevins", "Username" : "Wbevins", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13024, "Middle Name" : "Casey", "Grade" : 9, "Status" : "Active", "Birthdate" : "4/13/2001", "Graduation Year" : 2020 }
{ "_id" : ObjectId("64069c6f11cd57574309cffb"), "SIS ID" : 13020, "School SIS ID" : 10001, "First Name" : "Rogelio", "Last Name" : "Cazares", "Username" : "Rcazares", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13025, "Middle Name" : "Hal", "Grade" : 10, "Status" : "Active", "Birthdate" : "2/1/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cffd"), "SIS ID" : 13022, "School SIS ID" : 10001, "First Name" : "Williams", "Last Name" : "Bisson", "Username" : "Wbisson", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13027, "Middle Name" : "Roxanne", "Grade" : 9, "Status" : "Active", "Birthdate" : "3/6/2001", "Graduation Year" : 2020 }
{ "_id" : ObjectId("64069c6f11cd57574309cffe"), "SIS ID" : 13002, "School SIS ID" : 10001, "First Name" : "Beulah", "Last Name" : "McMillan", "Username" : "BMcMillan", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13002, "Middle Name" : "Beulah", "Grade" : 9, "Status" : "Active", "Birthdate" : "5/17/1998", "Graduation Year" : 2017 }
```

Con está consulta le decimos que se salte un número de registros en concreto.

Con limit se quiere que el cursor(limite de resultados devueltos) devuelva solo el número de registros indicados

```
> db.test2.find().limit(1).skip(2);
{ "_id" : ObjectId("64069c6f11cd57574309cff6"), "SIS ID" : 13005, "School SIS ID" : 10001, "First Name" : "Erna", "Last Name" : "Parker", "Username" : "EParker", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13005, "Middle Name" : "Troy", "Grade" : 11, "Status" : "Active", "Birthdate" : "5/17/1998", "Graduation Year" : 2017 }
>
```

Esta consulta filtra solo los datos que contengan de un punto a otro



```
> db.test2.find({"Grade":{"$in": [9,10]}));
{ "_id" : ObjectId("64069c6f11cd57574309cff4"), "SIS ID" : 13001, "School SIS ID" : 10001, "First Name" : "Ora", "Last Name" : "Klein", "Username" : "OKlein", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13001, "Middle Name" : "Christopher", "Grade" : 9, "Status" : "Active", "Birthdate" : "4/2/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cff5"), "SIS ID" : 13006, "School SIS ID" : 10001, "First Name" : "Sherry", "Last Name" : "Santana", "Username" : "SSantana", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13006, "Middle Name" : "Thomas", "Grade" : 10, "Status" : "Active", "Birthdate" : "8/22/1999", "Graduation Year" : 2018 }
{ "_id" : ObjectId("64069c6f11cd57574309cff8"), "SIS ID" : 13017, "School SIS ID" : 10001, "First Name" : "Maribel", "Last Name" : "Parsons", "Username" : "Mparsons", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13022, "Middle Name" : "Delmar", "Grade" : 10, "Status" : "Active", "Birthdate" : "3/10/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cffa"), "SIS ID" : 13019, "School SIS ID" : 10001, "First Name" : "Wilfred", "Last Name" : "Bevins", "Username" : "Wbevins", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13024, "Middle Name" : "Casey", "Grade" : 9, "Status" : "Active", "Birthdate" : "4/13/2001", "Graduation Year" : 2020 }
{ "_id" : ObjectId("64069c6f11cd57574309cffb"), "SIS ID" : 13020, "School SIS ID" : 10001, "First Name" : "Rogelio", "Last Name" : "Cazares", "Username" : "Rcazares", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13025, "Middle Name" : "Hal", "Grade" : 10, "Status" : "Active", "Birthdate" : "2/1/2000", "Graduation Year" : 2019 }
{ "_id" : ObjectId("64069c6f11cd57574309cffd"), "SIS ID" : 13022, "School SIS ID" : 10001, "First Name" : "Williams", "Last Name" : "Bisson", "Username" : "Wbisson", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13027, "Middle Name" : "Roxanne", "Grade" : 9, "Status" : "Active", "Birthdate" : "3/6/2001", "Graduation Year" : 2020 }
{ "_id" : ObjectId("64069c6f11cd57574309cffe"), "SIS ID" : 13002, "School SIS ID" : 10001, "First Name" : "Beulah", "Last Name" : "McMillan", "Username" : "BMcMillan", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13002, "Middle Name" : "Beulah", "Grade" : 9, "Status" : "Active", "Birthdate" : "4/13/2001", "Graduation Year" : 2020 }
```

Esta consulta filtra únicamente el dato que contenga un dato en específico

```
> db.test2.find({"SIS ID":13014});
{ "_id" : ObjectId("64069c6f11cd57574309d006"), "SIS ID" : 13014, "School SIS ID" : 10001, "First Name" : "Ricott", "Last Name" : "Cottle", "Username" : "Rcott", "Password" : "P@ssword", "State ID" : "WA", "Secondary Email" : "", "Student Number" : 13019, "Middle Name" : "Forrest", "Grade" : 10, "Status" : "Active", "Birthdate" : "8/9/2000", "Graduation Year" : 2019 }
>
gen: 338"
```

Esta consulta nos filtra todos los datos Username de las collections

```
> db.test2.find({}, {"Username":1});
{ "_id" : ObjectId("64069c6f11cd57574309cff4"), "Username" : "OKlein" }
{ "_id" : ObjectId("64069c6f11cd57574309cff5"), "Username" : "SSantana" }
{ "_id" : ObjectId("64069c6f11cd57574309cff6"), "Username" : "EParker" }
{ "_id" : ObjectId("64069c6f11cd57574309cff7"), "Username" : "FStark" }
{ "_id" : ObjectId("64069c6f11cd57574309cff8"), "Username" : "Mparsons" }
{ "_id" : ObjectId("64069c6f11cd57574309cff9"), "Username" : "Eballard" }
{ "_id" : ObjectId("64069c6f11cd57574309cffa"), "Username" : "Wbevins" }
{ "_id" : ObjectId("64069c6f11cd57574309cffb"), "Username" : "Rcazares" }
{ "_id" : ObjectId("64069c6f11cd57574309cffc"), "Username" : "Dmorrison" }
{ "_id" : ObjectId("64069c6f11cd57574309cffd"), "Username" : "Wbisson" }
{ "_id" : ObjectId("64069c6f11cd57574309cffe"), "Username" : "BMcMillan" }
{ "_id" : ObjectId("64069c6f11cd57574309cfff"), "Username" : "PHampton" }
{ "_id" : ObjectId("64069c6f11cd57574309d000"), "Username" : "DMatheson" }
{ "_id" : ObjectId("64069c6f11cd57574309d001"), "Username" : "LPratt" }
{ "_id" : ObjectId("64069c6f11cd57574309d002"), "Username" : "MThomas" }
{ "_id" : ObjectId("64069c6f11cd57574309d003"), "Username" : "CMcCray" }
{ "_id" : ObjectId("64069c6f11cd57574309d004"), "Username" : "PBarlow" }
{ "_id" : ObjectId("64069c6f11cd57574309d005"), "Username" : "Fmarkley" }
{ "_id" : ObjectId("64069c6f11cd57574309d006"), "Username" : "Rcottle" }
{ "_id" : ObjectId("64069c6f11cd57574309d007"), "Username" : "RLees" }
```

Esto básicamente fue una contextualización de lo que es usar MongoDB, se pueden importar otro tipo de archivos al mismo. En este caso manejamos archivo json y se pudo demostrar que las bases de datos no relacionales se manejan de una forma sencilla y agradable para los desarrolladores.

Conclusión de trabajo con Mongo

-  
-  
-  
--