Saulo Gabriel López Antonio

**data-analysis-gdl-20-04**

Proyectos de sesiones

# Sesión 1: Fundamentos de SQL

1. ¿Cuál es el nombre de los empleados con el puesto 4?

select \*

from empleado

where id\_puesto = 4;

1. Qué puestos tienen salarios mayor a 10000

select \*

from puesto

where salario > 10000;

1. ¿Que articulos tienen precio mayor a $1000 y un iva mayor a 100

select \*

from articulo

where precio > 1000

and iva > 100;

1. ¿Qué ventas incluyen los articulos 135 o 963 y fueron hechas por los empleados 835 o 369

select \*

from venta

where id\_articulo in (135,963)

and id\_empleado in (835,369);

# Proyecto Sesion 1

1. Dentro del mismo servidor de bases de datos, conéctate al esquema classicmodels.

use classicmodels;

1. Dentro de la tabla employees, obtén el apellido de todos los empleados.

select

lastName

from employees;

1. Dentro de la tabla employees, obtén el apellido, nombre y puesto de todos los empleados

select

lastName,

firstName,

jobTitle

from employees;

1. Dentro de la tabla employees, obtén todos los datos de cada empleado

select \*

from employees;

1. Dentro de la tabla employees, obtén el apellido, nombre y puesto de todos los empleados que tengan el puesto Sales Rep.

select

lastName,

firstName,

jobTitle

from employees

where jobTitle = "Sales Rep";

1. Dentro de la tabla employees, obtén el apellido, nombre, puesto y código de oficina de todos los empleados que tengan el puesto Sales Rep y código de oficina 1.

select

lastName,

firstName,

jobTitle,

officeCode

from employees

where jobTitle = "Sales Rep"

and officeCode = 1;

1. Dentro de la tabla employees, obtén el apellido, nombre y código de oficina de todos los empleados que tenga código de oficina 1, 2 o 3.

select

lastName,

firstName,

officeCode

from employees

where officeCode in (1,2,3);

1. Dentro de la tabla employees, obten el apellido, nombre y puesto de todos los empleados que tengan un puesto distinto a Sales Rep.

select

lastName,

firstName,

jobTitle

from employees

where jobTitle <> "Sales Rep";

1. Dentro de la tabla employees, obtén el apellido, nombre y código de oficina de todos los empleados cuyo código de oficina sea mayor a 5.

select

lastName,

firstName,

officeCode

from employees

where officeCode > 5;

1. Dentro de la tabla employees, obtén el apellido, nombre y código de oficina de todos los empleados cuyo cdigo de oficina sea menor o igual 4.

select

lastName,

firstName,

officeCode

from employees

where officeCode <= 4;

1. Dentro de la tabla customers, obtén el nombre, país y estado de todos los clientes cuyo país sea USA y cuyo estado sea CA.

select

customerName,

country,

state

from customers

where country = "USA"

AND state = "CA";

1. Dentro de la tabla customers, obtén el nombre, país, estado y límite de crédito de todos los clientes cuyo país sea, USA, cuyo estado sea CA y cuyo límite de crédito sea mayor a 100000.

select

customerName,

country,

state,

creditLimit

from customers

where country = "USA"

AND state = "CA"

AND creditLimit > 100000;

1. Dentro de la tabla customers, obtén el nombre y país de todos los clientes cuyo país sea USA o France.

select

customerName,

country

from customers

where country in ( "USA" , "France")

;

1. Dentro de la tabla customers, obtén el nombre, pas y límite de crédito de todos los clientes cuyo país sea USA o France y cuyo límite de crédito sea mayor a 100000. Para este ejercicio ten cuidado con los paréntesis.

select

customerName,

country,

creditLimit

from customers

where country in ( "USA" , "France")

AND creditLimit > 100000;

1. Dentro de la tabla offices, obtén el código de la oficina, ciudad, teléfono y país de aquellas oficinas que se encuentren en USA o France.

select

officeCode,

city,

phone,

country

from offices;

1. Dentro de la tabla offices, obtén el código de la oficina, ciudad, teléfono y país de aquellas oficinas que no se encuentren en USA o France.

select

officeCode,

city,

phone,

country

from offices

where country <> "USA"

and country <> "France" ;

1. Dentro de la tabla orders, obtén el número de orden, número de cliente, estado y fecha de envío de todas las órdenes con el número 10165, 10287 o 10310.

select

orderNumber,

customerNumber,

'status',

shippedDate

from orders;

1. Dentro de la tabla customers, obtén el apellido y nombre de cada cliente y ordena los resultados por apellido de forma ascendente.

select

contactLastName,

contactFirstName

from customers

order by contactLastName asc;

1. Dentro de la tabla customers, obtén el apellido y nombre de cada cliente y ordena los resultados por apellido de forma descendente

select

contactLastName,

contactFirstName

from customers

order by contactLastName desc;

1. Dentro de la tabla customers, obtén el apellido y nombre de cada cliente y ordena los resultados por apellido de forma descendente y luego por nombre de forma ascendente

select

contactLastName,

contactFirstName

from customers

order by contactLastName desc, contactFirstName asc;

1. Dentro de la tabla customers, obtén el número de cliente, nombre de cliente y el límite de crédito de los cinco clientes con el límite de crédito más alto (top 5).

select

customerNumber,

customerName,

creditlimit

from customers

order by creditlimit desc

limit 5;

1. Dentro de la tabla customers, obtén el número de cliente, nombre de cliente y el límite de crédito de los cinco clientes con el límite de crédito más bajo

select

customerNumber,

customerName,

creditlimit

from customers

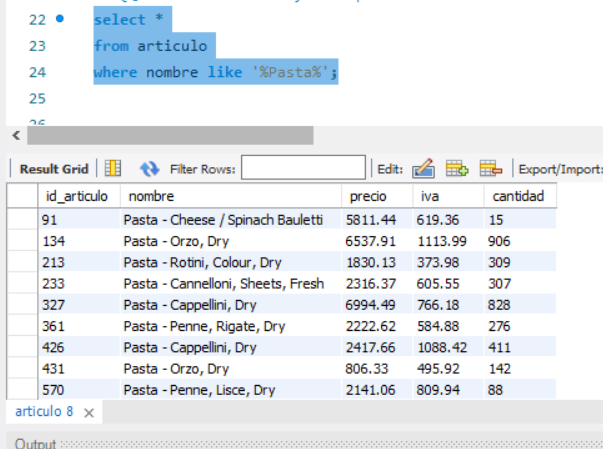
order by creditlimit

limit 5;

# Sesión 2: Agrupaciones y subconsultas

## Reto 1

* **¿Qué artículos incluyen la palabra Pasta en su nombre?**

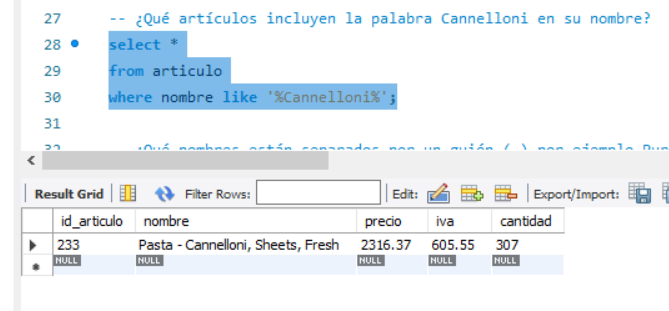
****

* **¿Qué artículos incluyen la palabra Cannelloni en su nombre?**

**select \***

**from articulo**

**where nombre like '%Cannelloni%';**

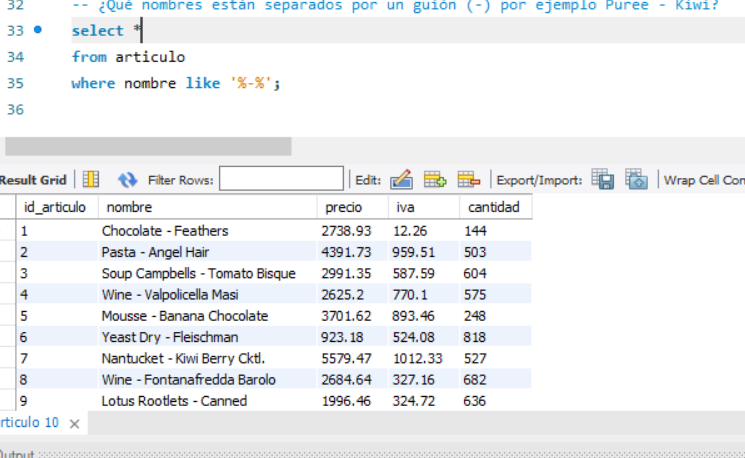
****

* **¿Qué nombres están separados por un guión (-) por ejemplo Puree - Kiwi?**

**select \***

**from articulo**

**where nombre like '%-%';**

****

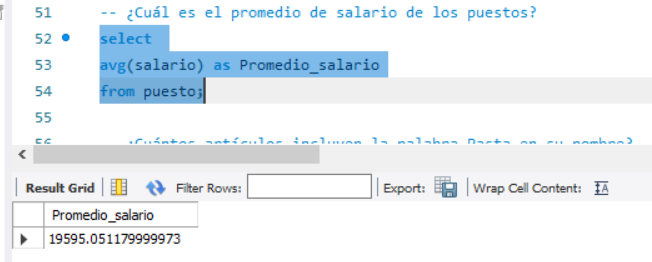
## Reto 2

* **¿Cuál es el promedio de salario de los puestos?**

**select**

**avg(salario) as Promedio\_salario**

**from puesto;**

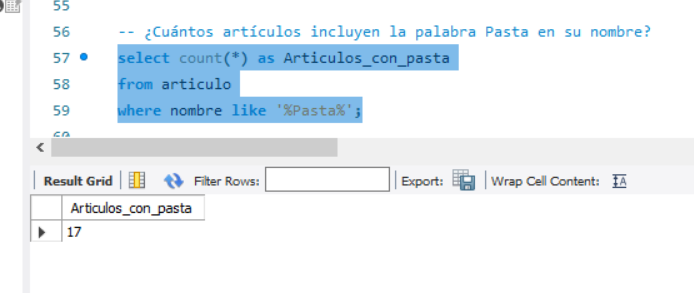
****

* **¿Cuántos artículos incluyen la palabra Pasta en su nombre?**

**select count(\*) as Articulos\_con\_pasta**

**from articulo**

**where nombre like '%Pasta%';**

****

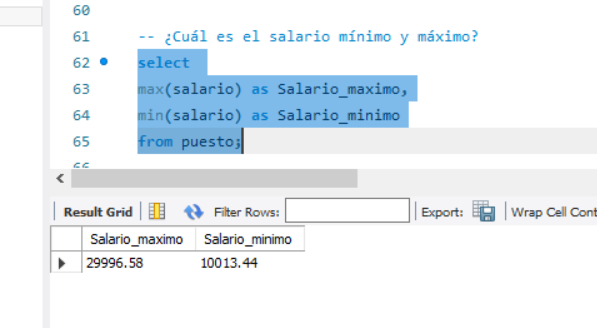
* **¿Cuál es el salario mínimo y máximo?**

**select**

**max(salario) as Salario\_maximo,**

**min(salario) as Salario\_minimo**

**from puesto;**

****

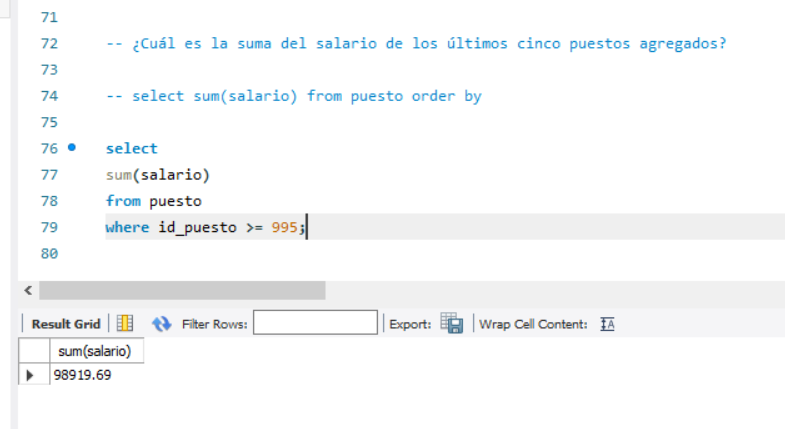
* **¿Cuál es la suma del salario de los últimos cinco puestos agregados?**

**select**

**sum(salario)**

**from puesto**

**where id\_puesto >= 995;**



# Reto 3

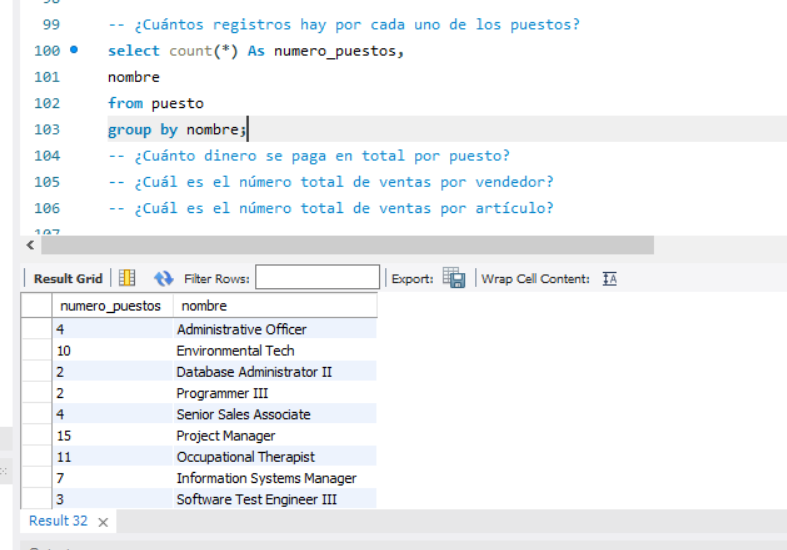
* ¿Cuántos registros hay por cada uno de los puestos?

select count(\*) As numero\_puestos,

nombre

from puesto

group by nombre;



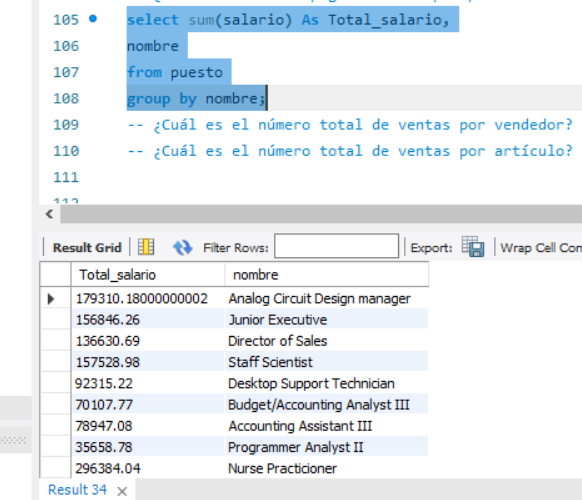
* ¿Cuánto dinero se paga en total por puesto?

select sum(salario) As Total\_salario,

nombre

from puesto

group by nombre;

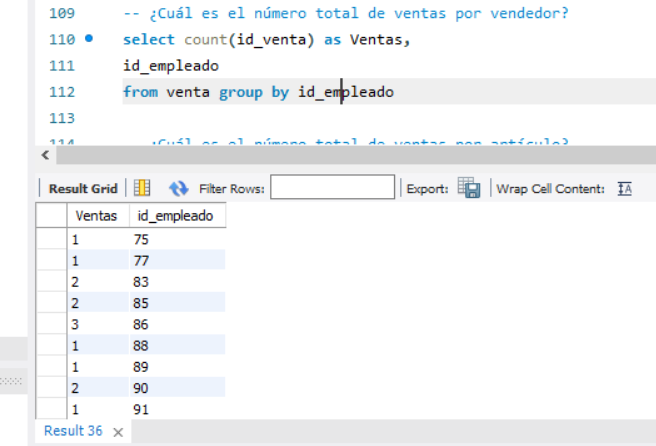


* ¿Cuál es el número total de ventas por vendedor?

select count(id\_venta) as Ventas,

id\_empleado

from venta group by id\_empleado

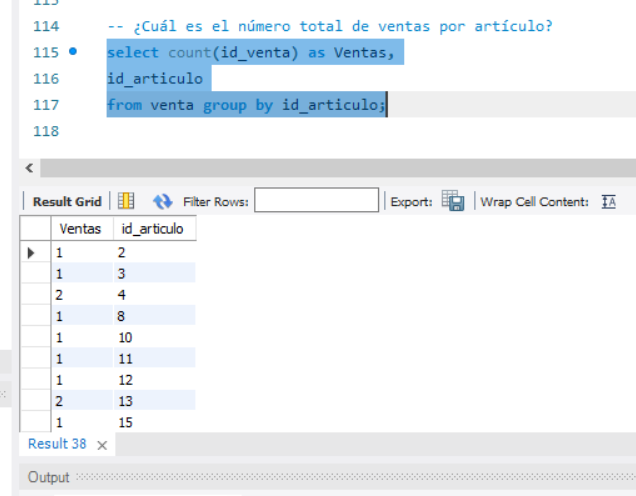


* ¿Cuál es el número total de ventas por artículo?

select count(id\_venta) as Ventas,

id\_articulo

from venta group by id\_articulo;



## Reto 4

* ¿Cuál es el nombre de los empleados cuyo sueldo es menor a $10,000?

select nombre

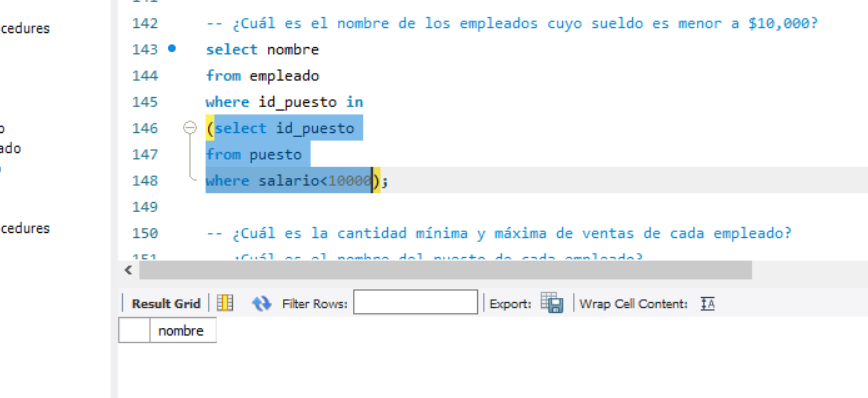
from empleado

where id\_puesto in

(select id\_puesto

from puesto

where salario<10000);



* ¿Cuál es la cantidad mínima y máxima de ventas de cada empleado?

select id\_empleado,

max(total\_ventas),

min(total\_ventas)

from

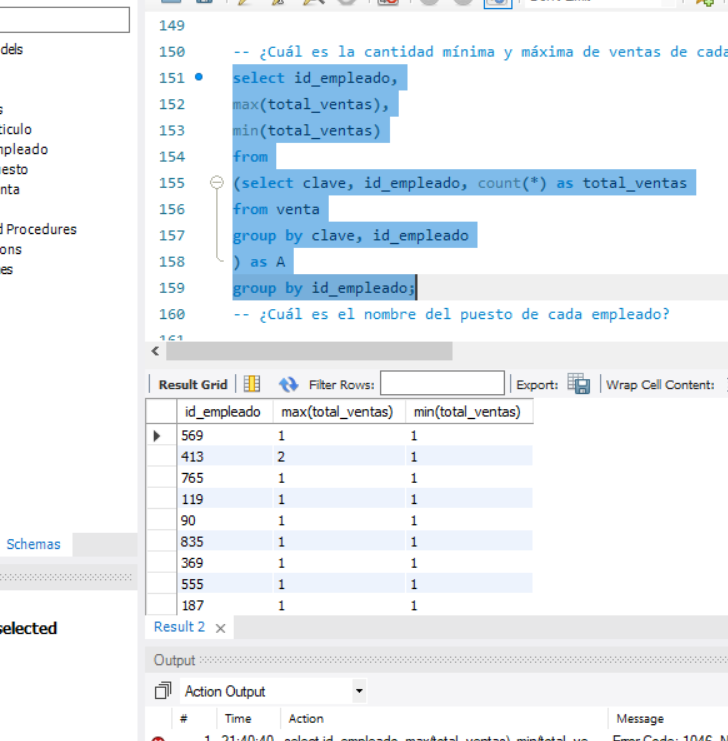
(select clave, id\_empleado, count(\*) as total\_ventas

from venta

group by clave, id\_empleado

) as A

group by id\_empleado;



* ¿Cuál es el nombre del puesto de cada empleado?

select

nombre,

apellido\_paterno,

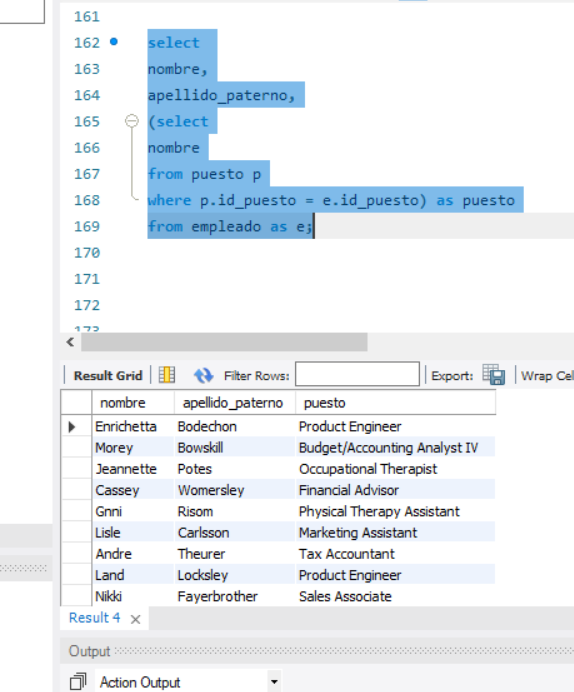
(select

nombre

from puesto p

where p.id\_puesto = e.id\_puesto) as puesto

from empleado as e;



# PROYECTO SESION 2

1. Dentro de la tabla employees, obten el número de empleado, apellido y nombre de todos los empleados cuyo nombre empiece con a.

select

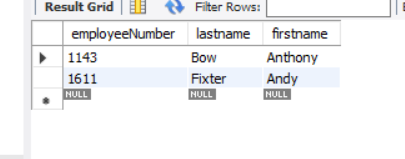
employeeNumber,

lastname,

firstname

from employees

where firstname like 'A%';



1. Dentro de la tabla employees, obten el número de empleado, apellido y nombre de todos los empleados cuyo nombre termina con on.

select

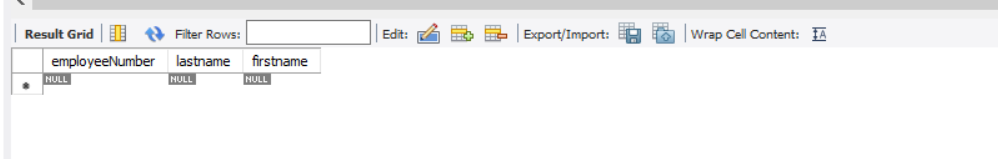
employeeNumber,

lastname,

firstname

from employees

where firstname like '%on';



1. Dentro de la tabla employees, obten el número de empleado, apellido y nombre de todos los empleados cuyo nombre incluye la cadena on.

select

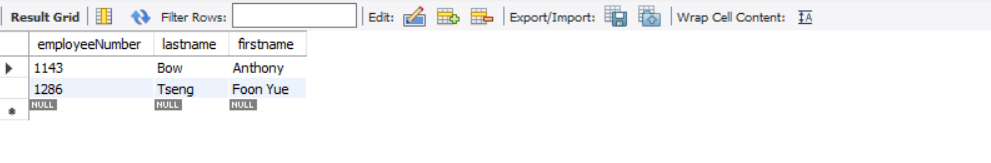
employeeNumber,

lastname,

firstname

from employees

where firstname like '%on%';



1. Dentro de la tabla employees, obten el número de empleado, apellido y nombre de todos los empleados cuyos nombres tienen tres letras e inician con T y finalizan con m.

select

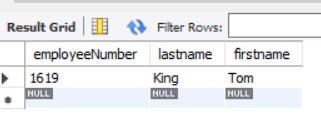
employeeNumber,

lastname,

firstname

from employees

where firstname like '%T\_m%';



1. Dentro de la tabla employees, obten el número de empleado, apellido y nombre de todos los empleados cuyo nombre no inicia con B.

select

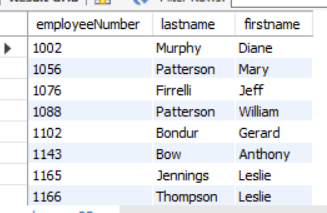
employeeNumber,

lastname,

firstname

from employees

where firstname not like 'B%';



1. Dentro de la tabla products, obten el código de producto y nombre de los productos cuyo código incluye la cadena \_20.

select

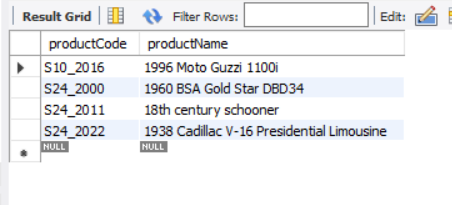
productCode,

productName

from

products

where productCode like'\_\_\_\_20%'



1. Dentro de la tabla orderdetails, obten el total de cada orden.

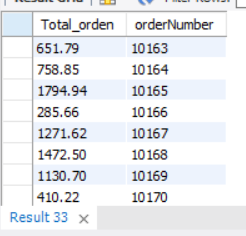
select

sum(priceEach) as Total\_orden,

orderNumber

from orderdetails

group by orderNumber;



1. Dentro de la tabla orders obten el número de órdenes por año.

select

count(\*) as ordenes,

year(orderDate) as 'year'

from orders

group by year(orderDate)



1. Obten el apellido y nombre de los empleados cuya oficina está ubicada en USA.

select

lastName,

firstName

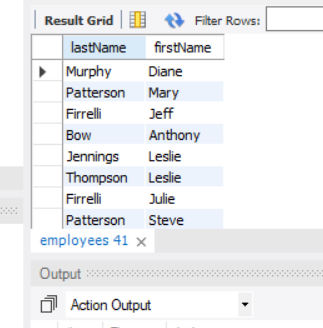
from employees

where officeCode in

(select officeCode

from offices

where country = 'USA');



1. Obten el número de cliente, número de cheque y cantidad del cliente que ha realizado el pago más alto.

select

customerNumber,

checkNumber,

amount

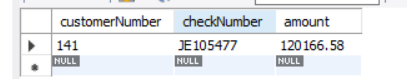
from payments

where amount in

(select

max(amount)

from payments);



1. Obten el número de cliente, número de cheque y cantidad de aquellos clientes cuyo pago es más alto que el promedio.

select

customerNumber,

checkNumber,

amount

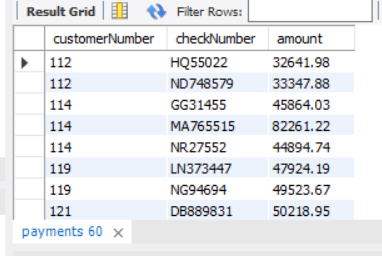
from payments

where amount >

(select

avg(amount)

from payments);



1. Obten el nombre de aquellos clientes que no han hecho ninguna orden.

select

contactLastName,

contactFirstName

from customers

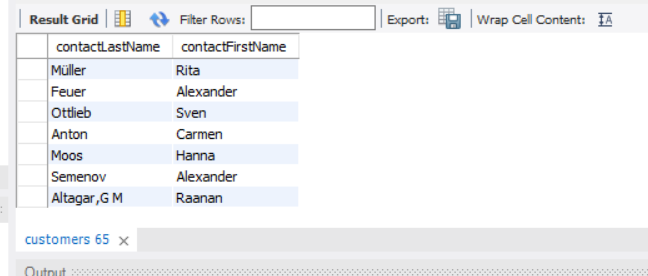
where customerNumber

not in

(select customerNumber

from orders

)



1. Obten el máximo, mínimo y promedio del número de productos en las órdenes de venta.

select

productCode,

max(quantityOrdered) as Max\_productos,

min(quantityOrdered) as Min\_productos,

avg(quantityOrdered) as Promedio\_productos

from orderdetails

group by productCode



select

D.Estado,

Sum(D.ordenes)

from

(select

C.state as Estado,

(select count(\*) as ordenes

from orders as B

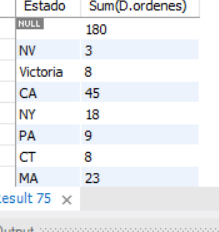
where B.customerNumber = C.customerNumber) as ordenes

from customers as C

) as D

group by

D.Estado;



# Sesión 3: Joins y Vistas

## Reto 1

¿Cuál es el nombre de los empleados que realizaron cada venta?

Reto 1

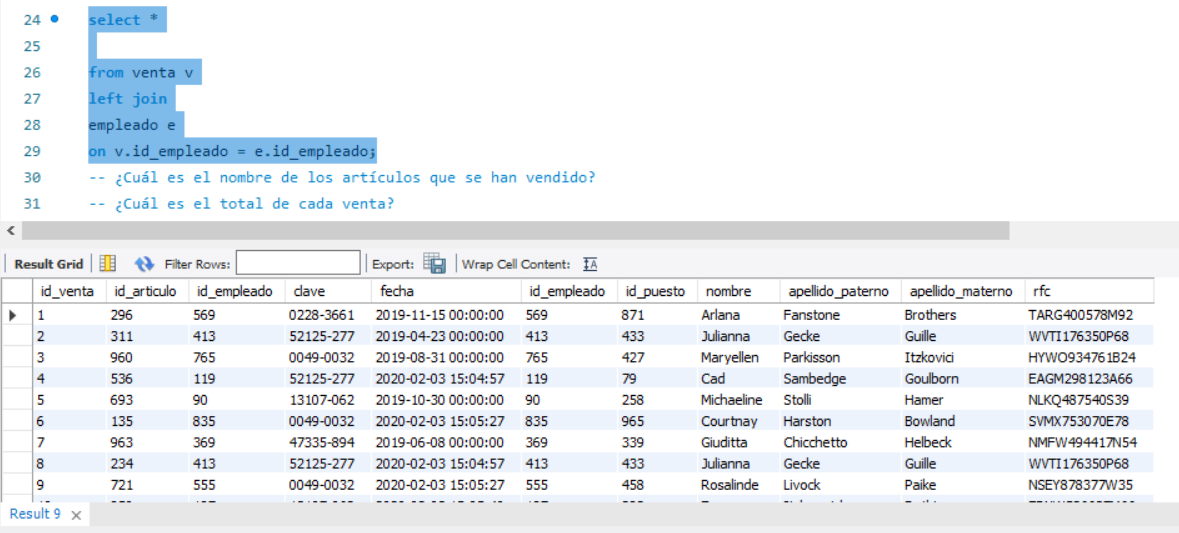
select \*

from venta v

left join

empleado e

on v.id\_empleado = e.id\_empleado;



¿Cuál es el nombre de los artículos que se han vendido?

SELECT

v.id\_venta,

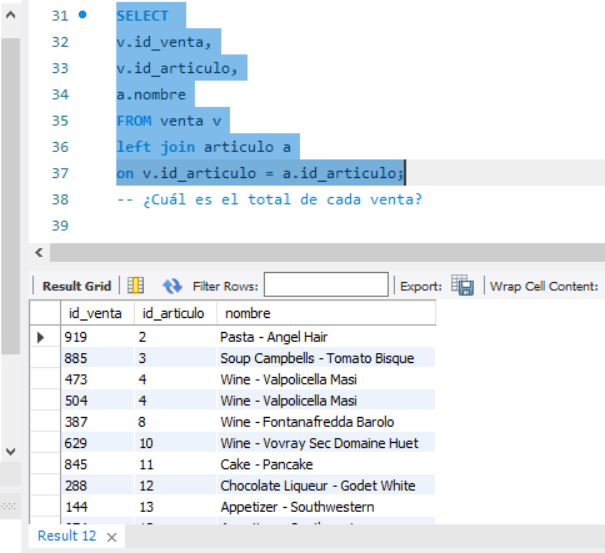
v.id\_articulo,

a.nombre

FROM venta v

left join articulo a

on v.id\_articulo = a.id\_articulo;



¿Cuál es el total de cada venta?

select

v.clave as venta,

sum(a.precio)

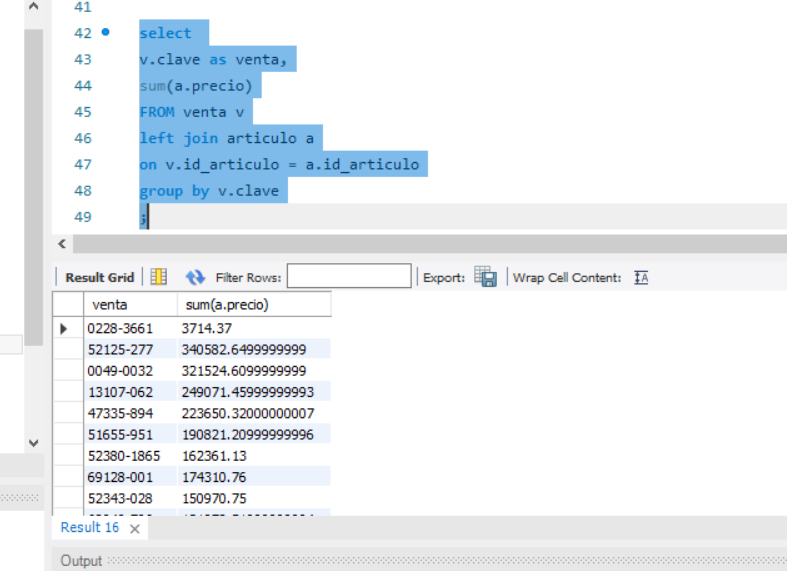
FROM venta v

left join articulo a

on v.id\_articulo = a.id\_articulo

group by v.clave

;



## Reto 2

create view SAUempleado as

(select

e.id\_puesto,

concat (e.nombre, ' ', e.apellido\_paterno, ' ', e.apellido\_materno) as empleado,

p.nombre as puesto

from empleado e left join

puesto p on e.id\_puesto = p.id\_puesto);

select \*

from SAUempleado;



* Saber qué artículos ha vendido cada empleado.

create view SAUarticulo\_ventas as

(select

v.id\_venta,

v.id\_articulo,

a.nombre as articulo,

concat (e.nombre, ' ', e.apellido\_paterno, ' ', e.apellido\_materno) as empleado

from

venta v

left join

articulo a

ON a.id\_articulo = v.id\_articulo

left join empleado e

ON v.id\_empleado = e.id\_empleado);

select

count(\*) as ventas,

articulo,

empleado

from

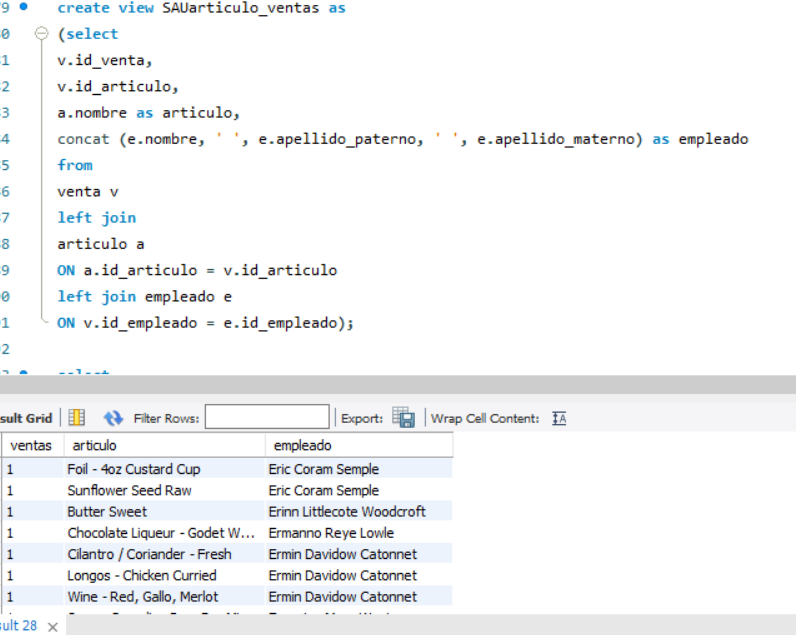
SAUarticulo\_ventas

group by

articulo,

empleado

order by empleado;



* Saber qué puesto ha tenido más ventas

create view SAUventas as

(select

v.id\_venta,

v.clave,

e.id\_empleado,

p.nombre as puesto

from

venta v

left join empleado e

ON v.id\_empleado = e.id\_empleado

left join puesto p

on e.id\_puesto = p.id\_puesto) ;

Select

puesto,

count(\*) as ventas

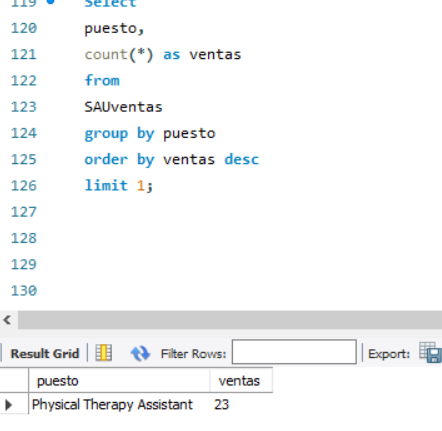
from

SAUventas

group by puesto

order by ventas desc

limit 1;



# PROYECTO SESION 3

-- Para estas consultas usa RIGHT JOIN

1) Obten el código de producto, nombre de producto y descripción de todos los productos.

select

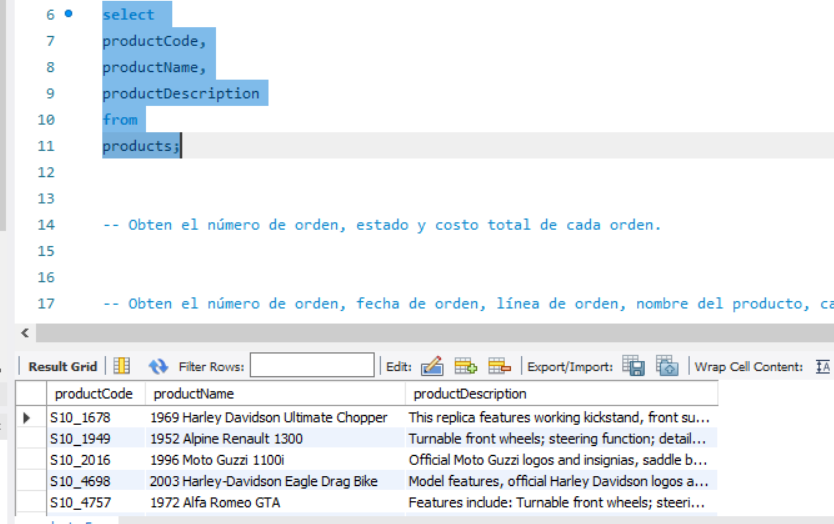
productCode,

productName,

productDescription

from

products;



2) Obten el número de orden, estado y costo total de cada orden.

select

o.orderNumber,

o.status,

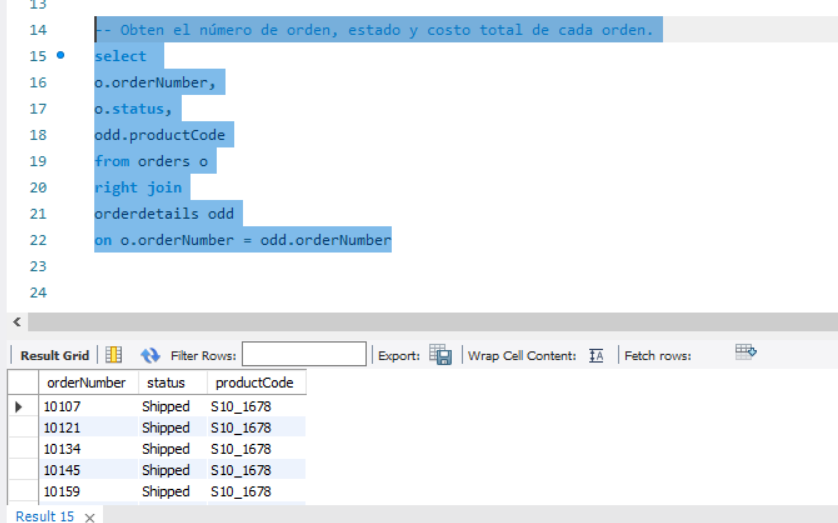
odd.productCode

from orders o

right join

orderdetails odd

on o.orderNumber = odd.orderNumber



3) Obten el número de orden, fecha de orden, línea de orden, nombre del producto, cantidad ordenada y precio de cada pieza que muestre los detalles de cada orden.

Select

O.orderNumber,

O.orderdate,

odd.orderLineNumber,

p.productName,

odd.quantityOrdered,

odd.priceEach

from

products p

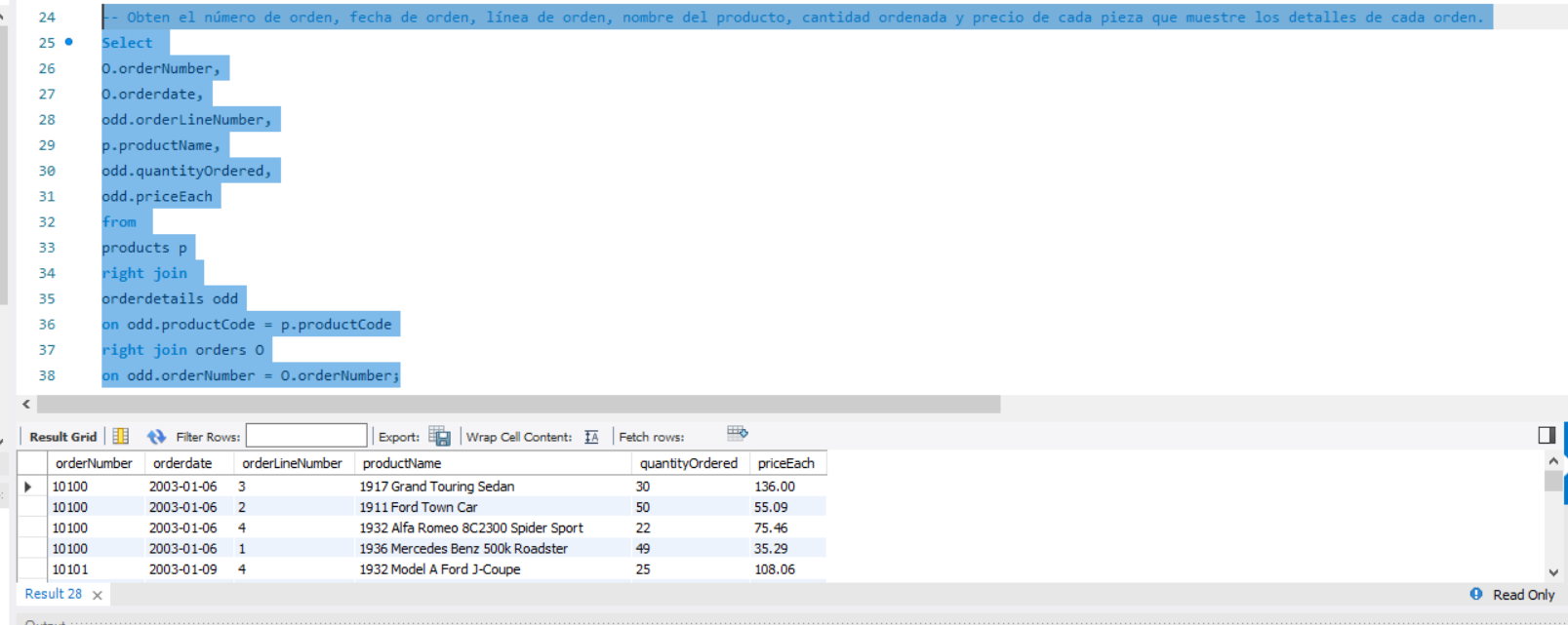
right join

orderdetails odd

on odd.productCode = p.productCode

right join orders O

on odd.orderNumber = O.orderNumber;



4) Obtén el número de orden, nombre del producto, el precio sugerido de fábrica (msrp) y precio de cada pieza.

select

O.orderNumber,

p.productname,

p.MSRP,

O.priceEach

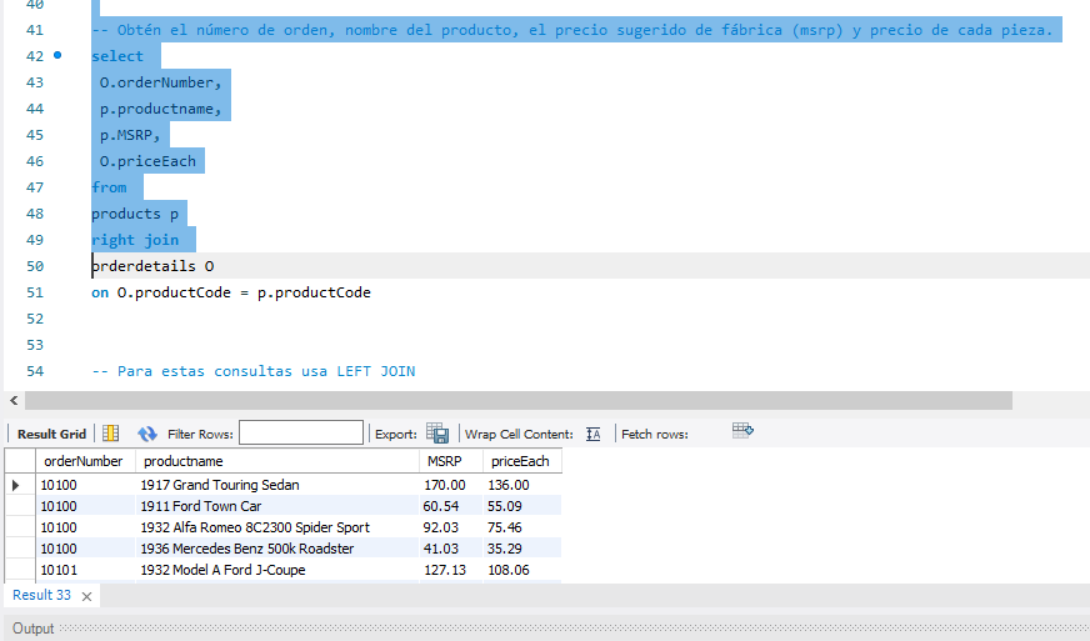
from

products p

right join

orderdetails O

on O.productCode = p.productCode



5) Obtén el número de cliente, nombre de cliente, número de orden y estado de cada cliente.

select

c.customernumber,

concat (c.contactFirstName,' ',contactLastName) as nombre\_cliente,

o.orderNumber,

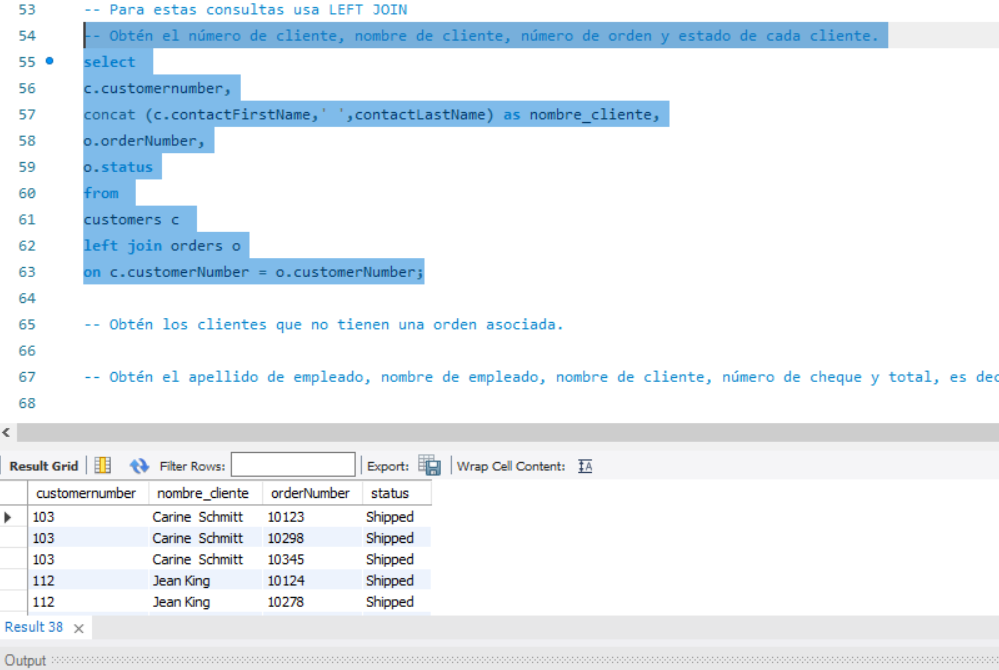
o.status

from

customers c

left join orders o

on c.customerNumber = o.customerNumber;



6) Obtén los clientes que no tienen una orden asociada.

select

c.customernumber,

concat (c.contactFirstName,' ',contactLastName) as nombre\_cliente,

o.orderNumber,

o.status

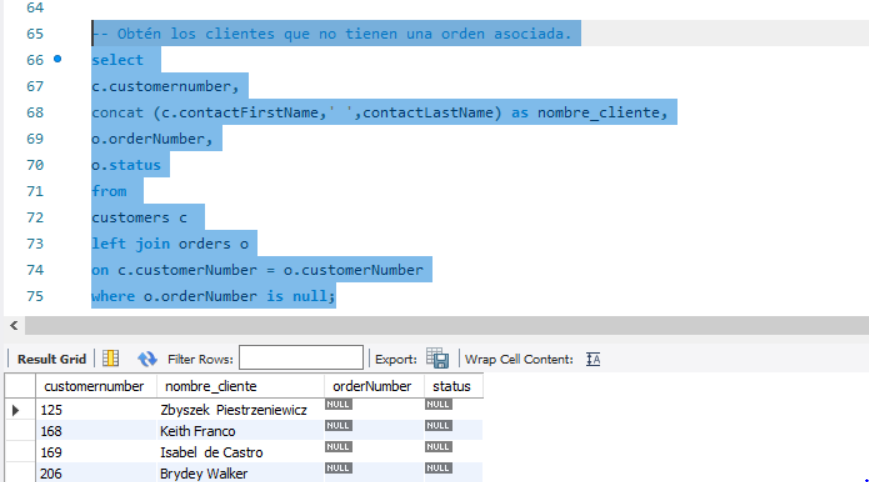
from

customers c

left join orders o

on c.customerNumber = o.customerNumber

where o.orderNumber is null;



7) Obtén el apellido de empleado, nombre de empleado, nombre de cliente, número de cheque y total, es decir, los clientes asociados a cada empleado.

select

e.lastName,

e.firstname,

c.customerNumber,

p.checkNumber,

p.amount

from customers c

left join

employees e

on c.salesRepEmployeeNumber = e.employeeNumber

left join

payments p

on c.customerNumber = p.customerNumber;



Para estas consultas usa RIGHT JOIN

Repite los ejercicios 5 a 7 usando RIGHT JOIN.

5.1.Obtén el número de cliente, nombre de cliente, número de orden y estado de cada cliente.

select

c.customernumber,

concat (c.contactFirstName,' ',contactLastName) as nombre\_cliente,

o.orderNumber,

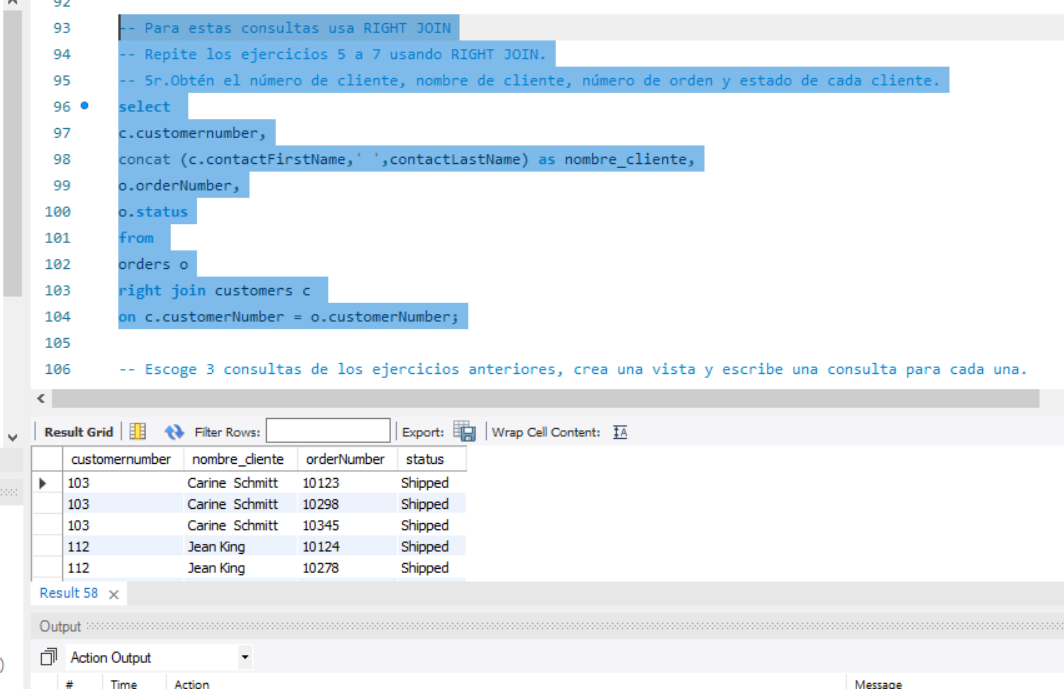
o.status

from

orders o

right join customers c

on c.customerNumber = o.customerNumber;



7.1. Obtén el apellido de empleado, nombre de empleado, nombre de cliente, número de cheque y total, es decir, los clientes asociados a cada empleado.

select

e.lastName,

e.firstname,

c.customerNumber,

p.checkNumber,

p.amount

from

payments p

right join

(employees e

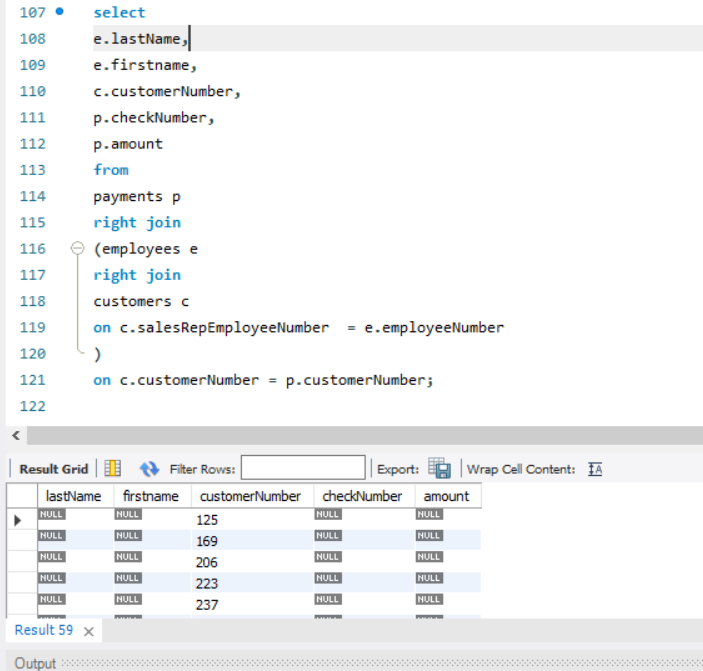
right join

customers c

on c.salesRepEmployeeNumber = e.employeeNumber

)

on c.customerNumber = p.customerNumber;



Escoge 3 consultas de los ejercicios anteriores, crea una vista y escribe una consulta para cada una.

create view SAUClientes\_sinorden as (

select

c.customernumber,

concat (c.contactFirstName,' ',contactLastName) as nombre\_cliente,

o.orderNumber,

o.status

from

customers c

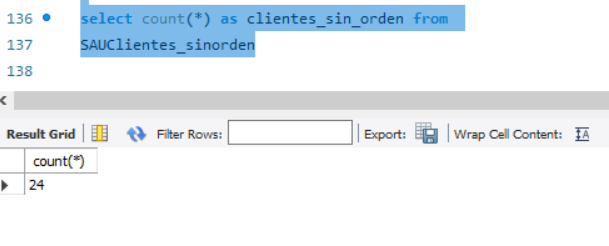
left join orders o

on c.customerNumber = o.customerNumber

where o.orderNumber is null);

select count(\*) as clientes\_sin\_orden from

SAUClientes\_sinorden



create view SAUClientes\_empleado as (

select

e.lastName,

e.firstname,

c.customerNumber,

p.checkNumber,

p.amount

from customers c

left join

employees e

on c.salesRepEmployeeNumber = e.employeeNumber

left join

payments p

on c.customerNumber = p.customerNumber);

select

concat(firstname,'',lastname) as Empleado,

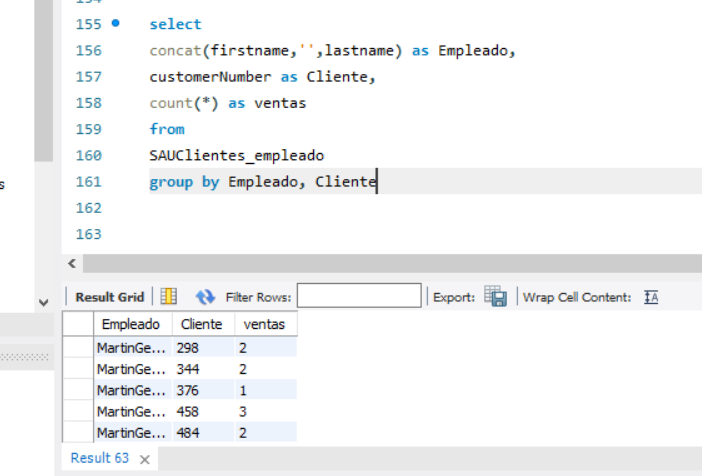
customerNumber as Cliente,

count(\*) as ventas

from

SAUClientes\_empleado

group by Empleado, Cliente



create view SAUProducto\_precio as (

select

O.orderNumber,

p.productname,

p.MSRP,

O.priceEach

from

products p

right join

orderdetails O

on O.productCode = p.productCode);

select

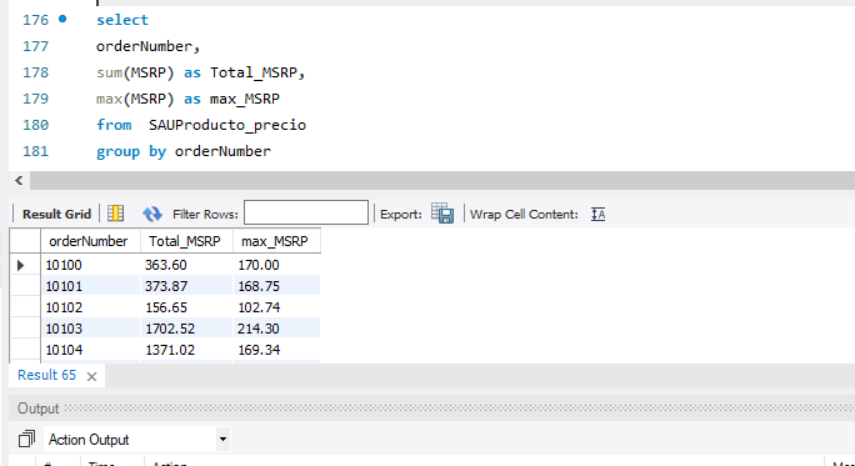
orderNumber,

sum(MSRP) as Total\_MSRP,

max(MSRP) as max\_MSRP

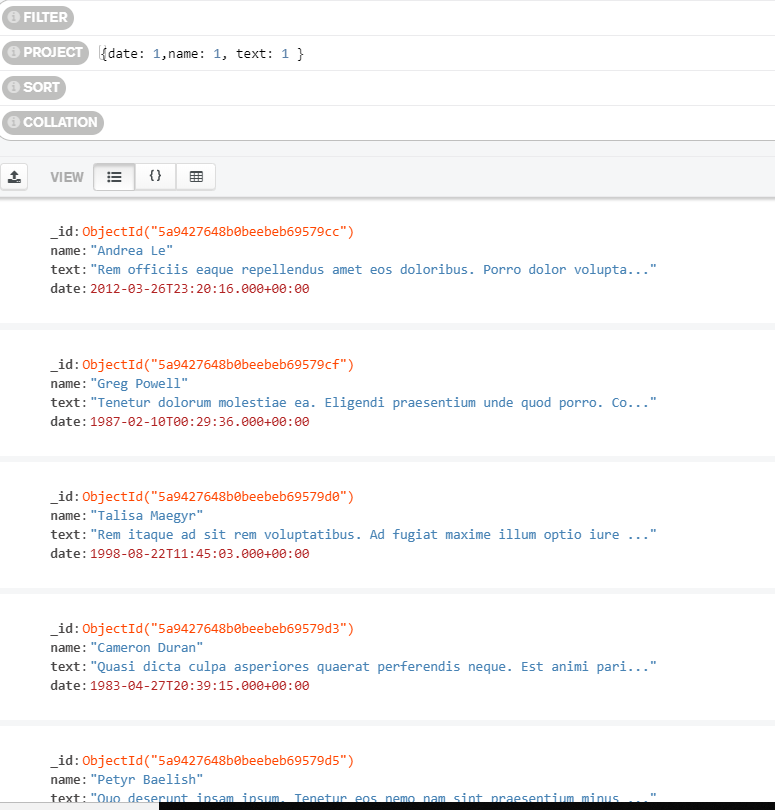
from SAUProducto\_precio

group by orderNumber

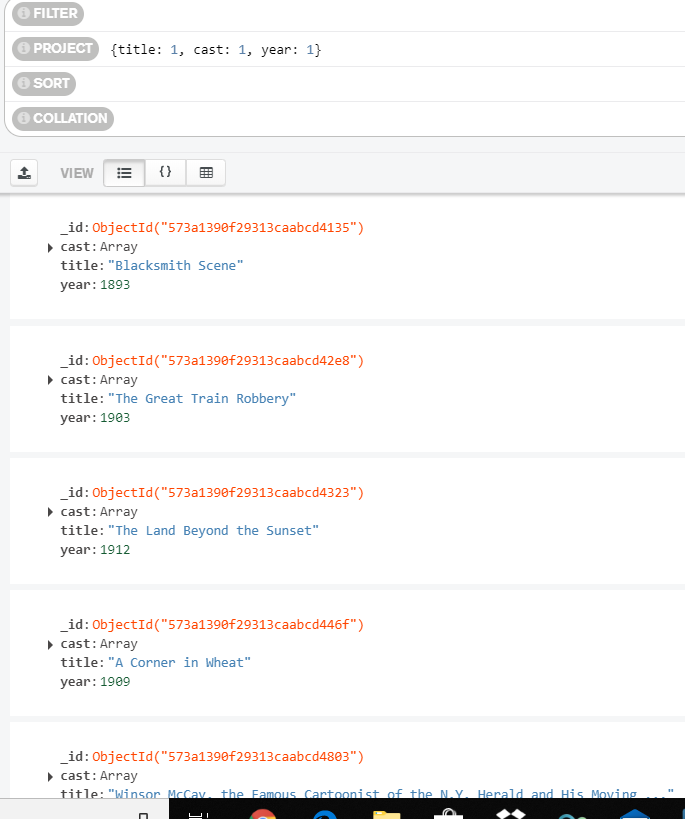


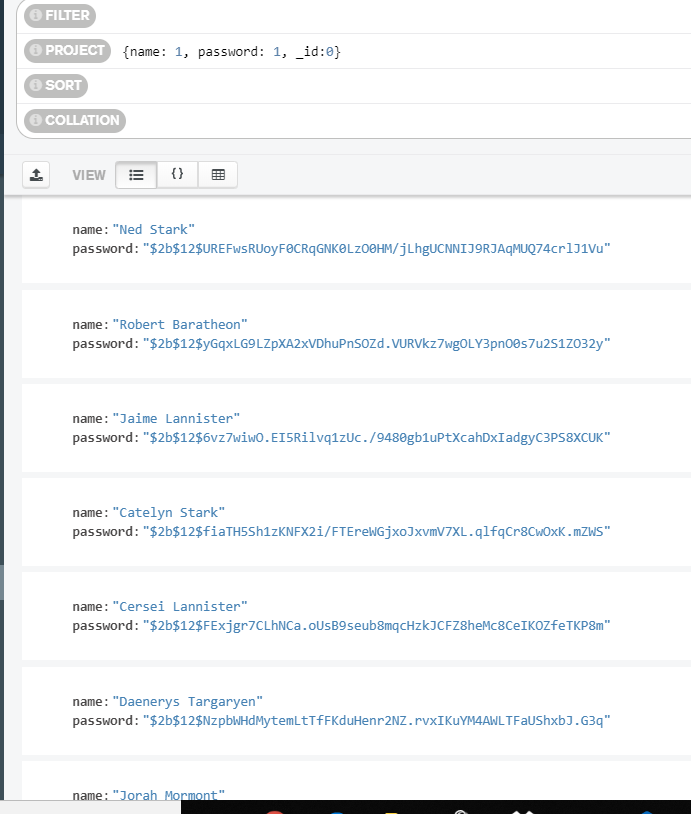
# Sesión 4: Fundamentos de MongoDB

* Fecha, nombre y texto de cada comentario.

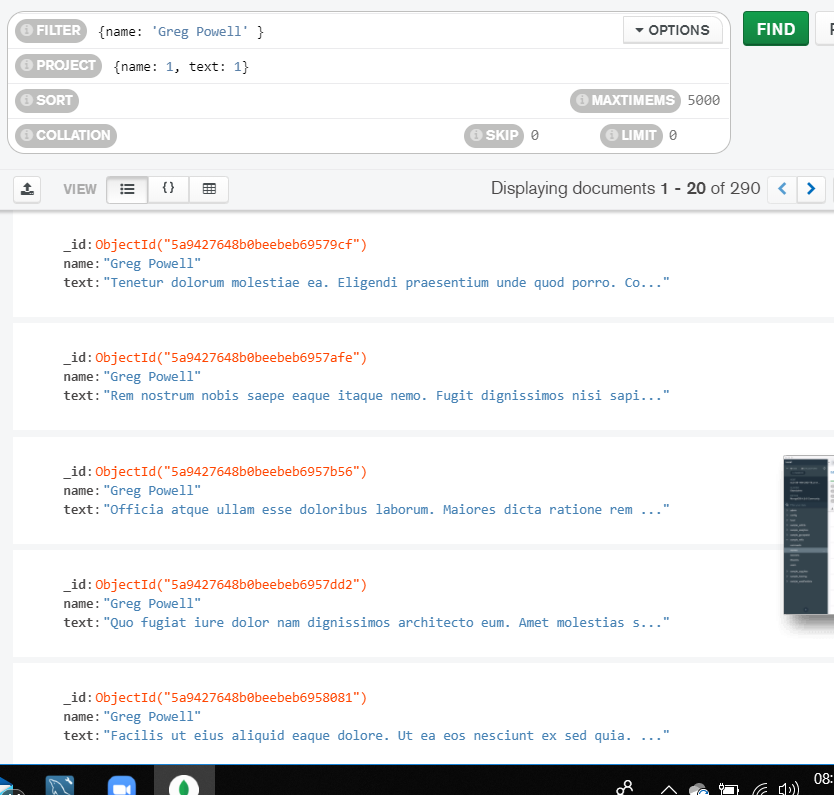


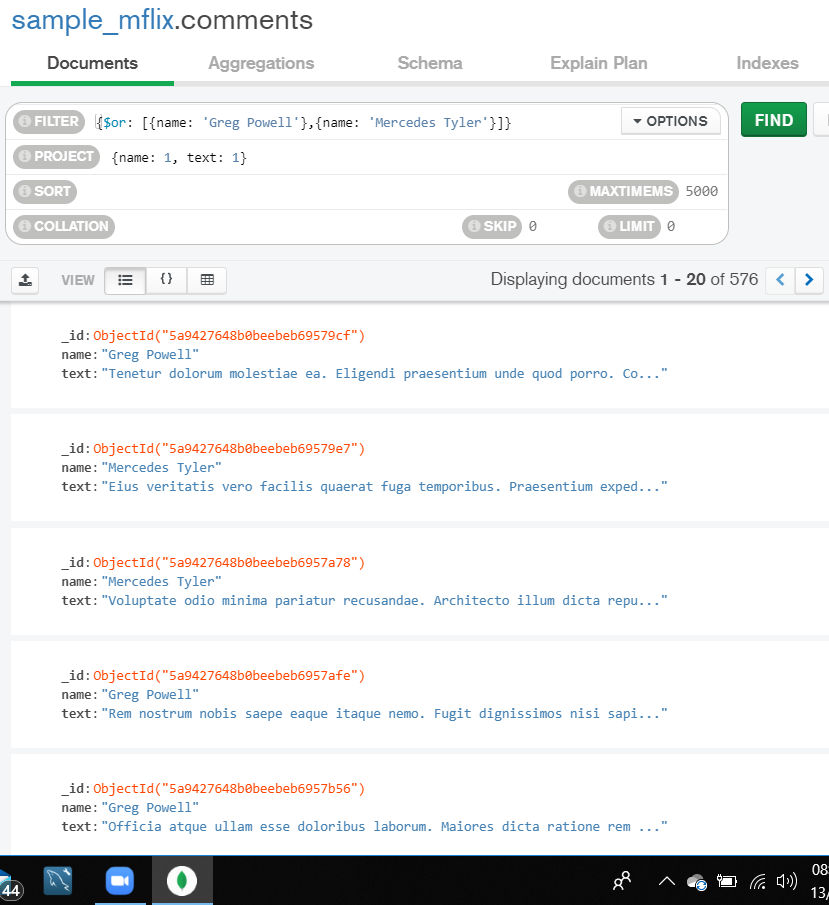
* Título, elenco y año de cada película.

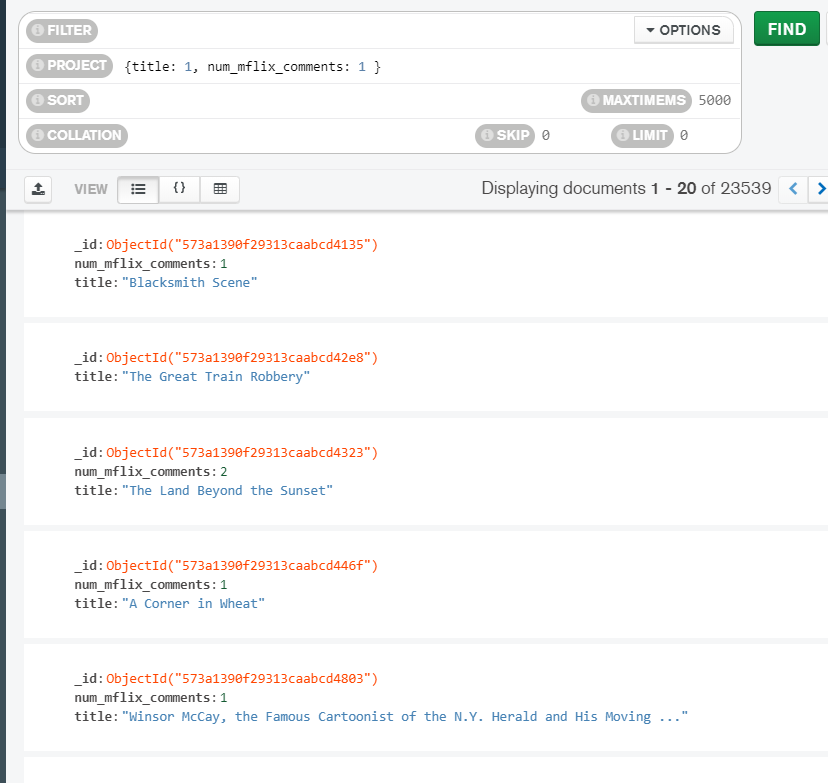
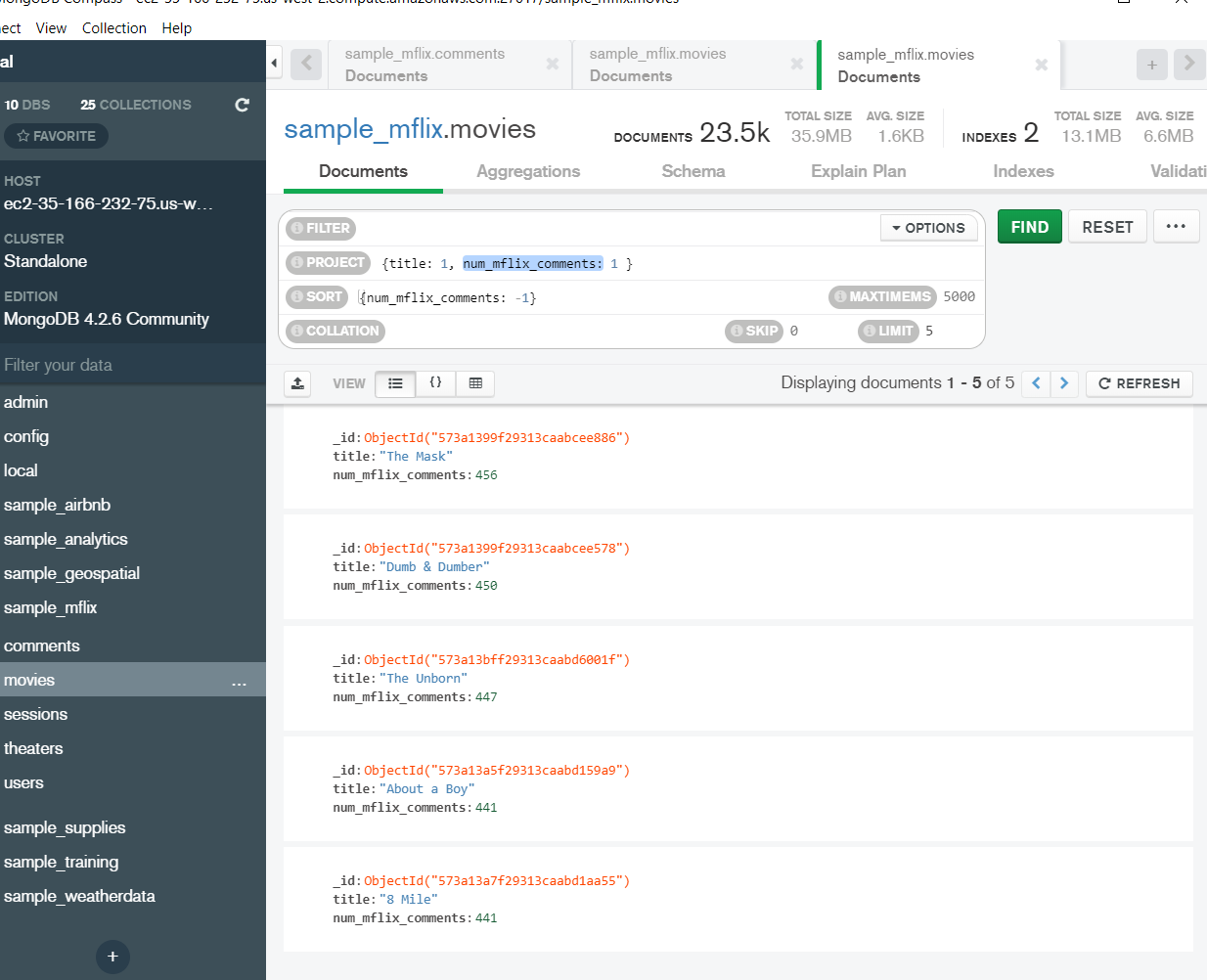


* Nombre y contraseña de cada usuario
* 

RETO 2

* ¿Qué comentarios ha hecho Greg Powell?
* 
* ¿Qué comentarios han hecho Greg Powell o Mercedes Tyler?



* ¿Cuál es el máximo número de comentarios en una película?
* 
* ¿Cuál es título de las cinco películas más comentadas?
* 

PROYECTO 3

* Obtén los datos de contacto de cada compañía.

{

project: {

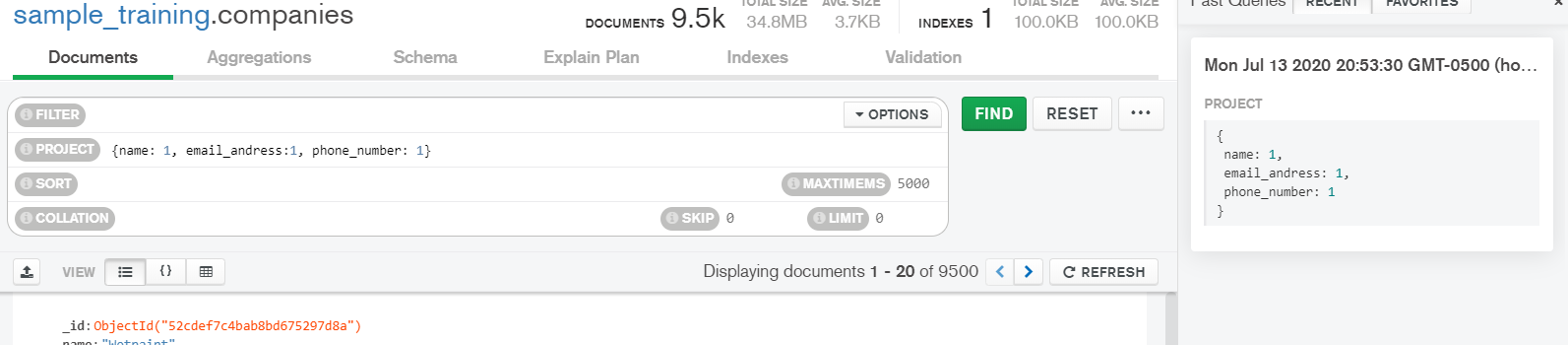
name: 1,

email\_andress: 1,

phone\_number: 1

}

}



* Obtén la fuente de cada tweet.

{

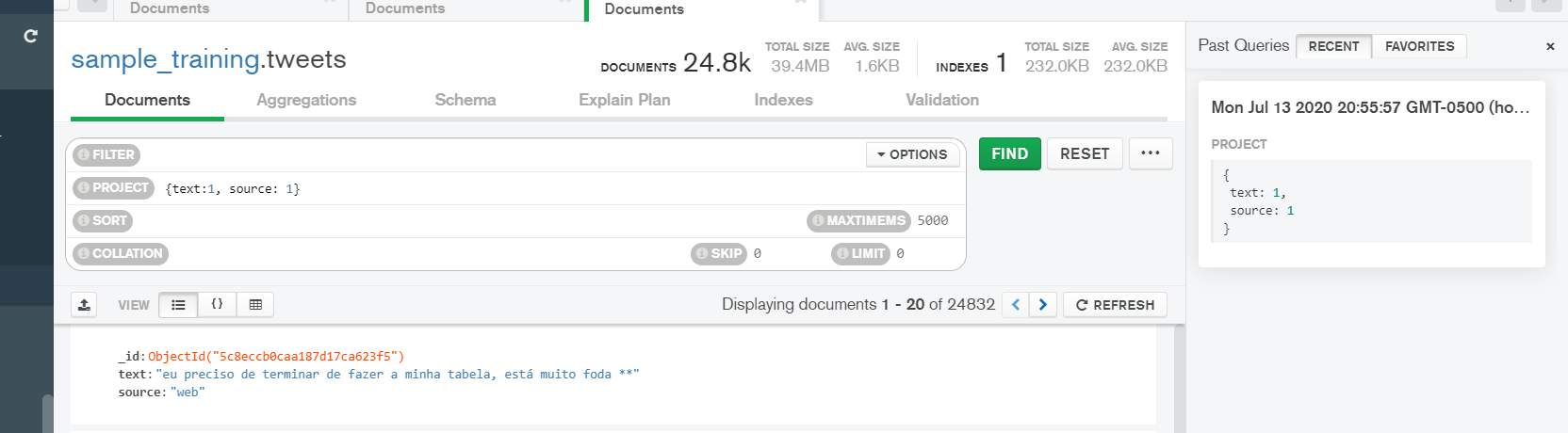
project: {

text: 1,

source: 1

}

}



* Obtén el nombre de todas las compañias fundadas en octubre.

{

filter: {

founded\_month: 10

},

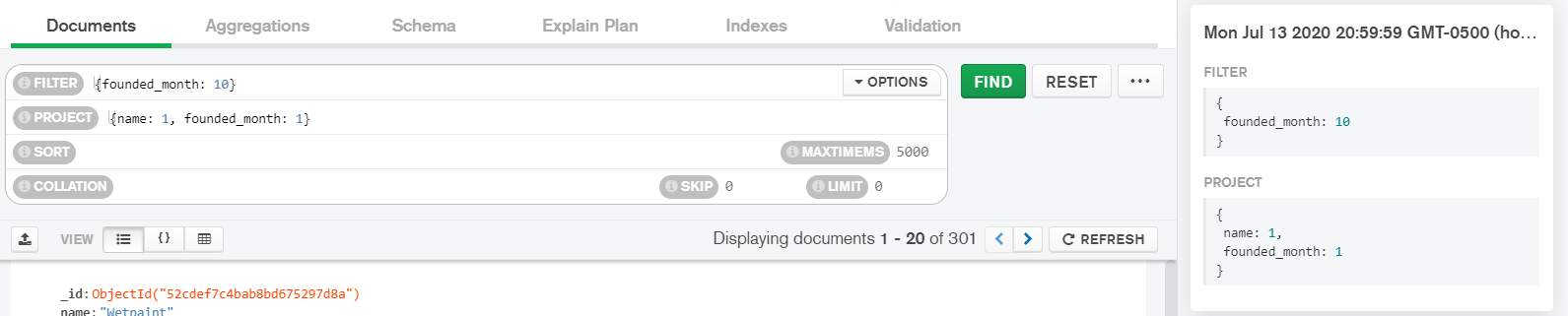
project: {

name: 1,

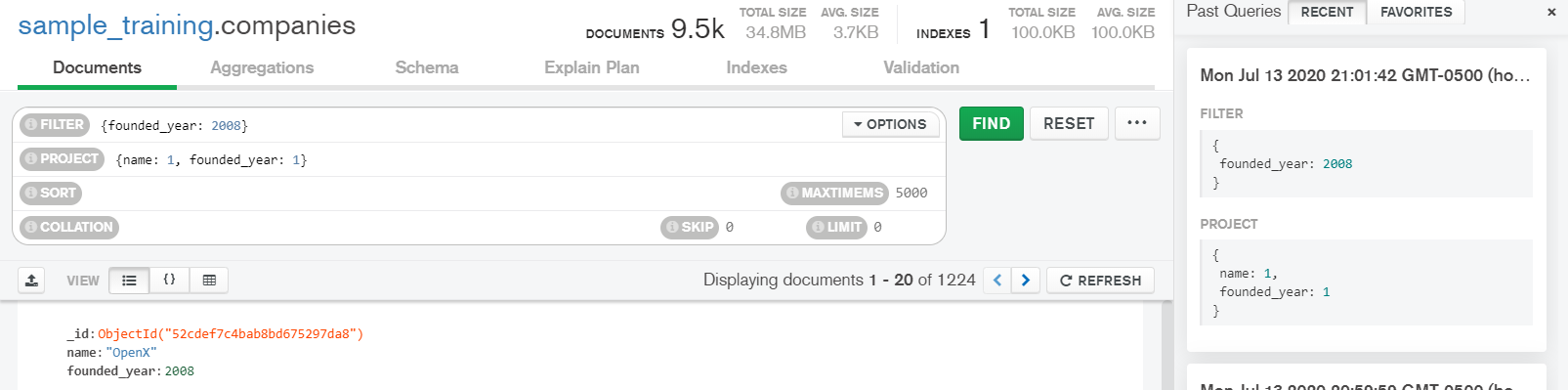
founded\_month: 1

}

}



* Obtén el nombre de todas las compañías fundadas en 2008.
* {
* filter: {
* founded\_year: 2008
* },
* project: {
* name: 1,
* founded\_year: 1
* }
* }



* Obtén todos los *post* del autor machine.

{

filter: {

author: 'machine'

},

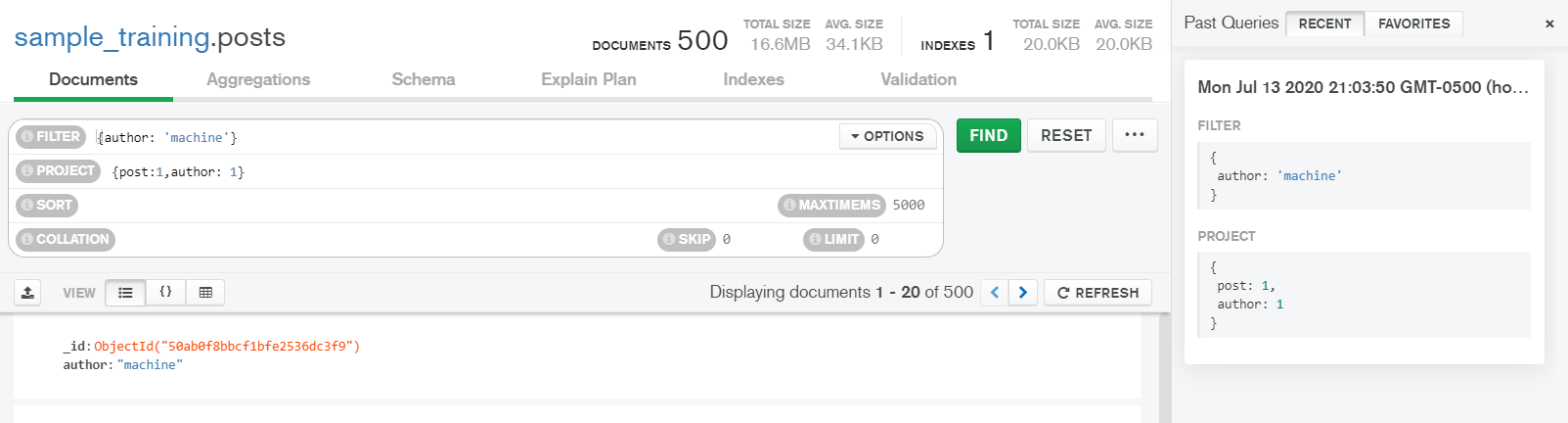
project: {

post: 1,

author: 1

}

}



* Obtén todos los tweets provenientes de la web.

{

filter: {

source: 'web'

},

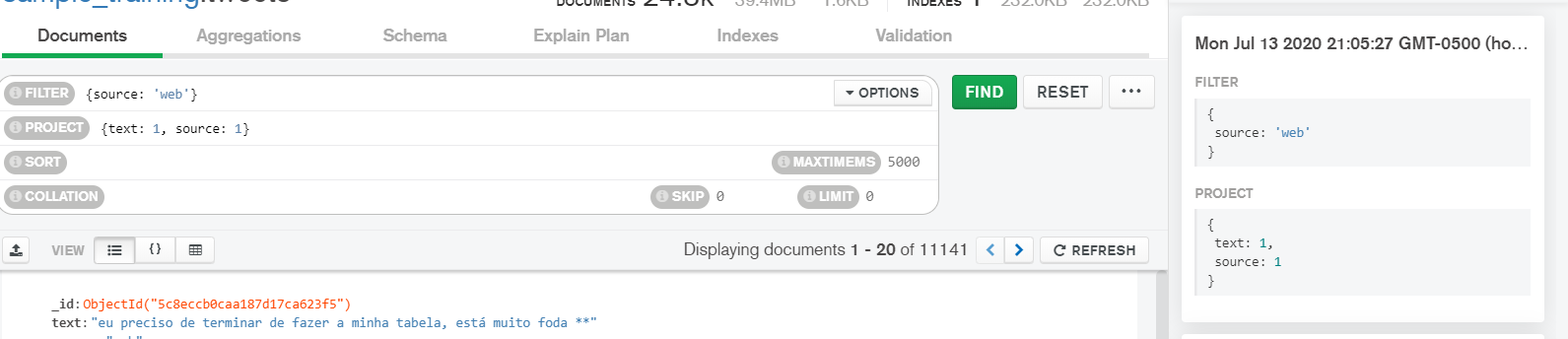
project: {

text: 1,

source: 1

}

}



* Obtén todas las compañías fundadas en octubre del 2008.

{

filter: {

founded\_month: 10,

founded\_year: 2008

},

project: {

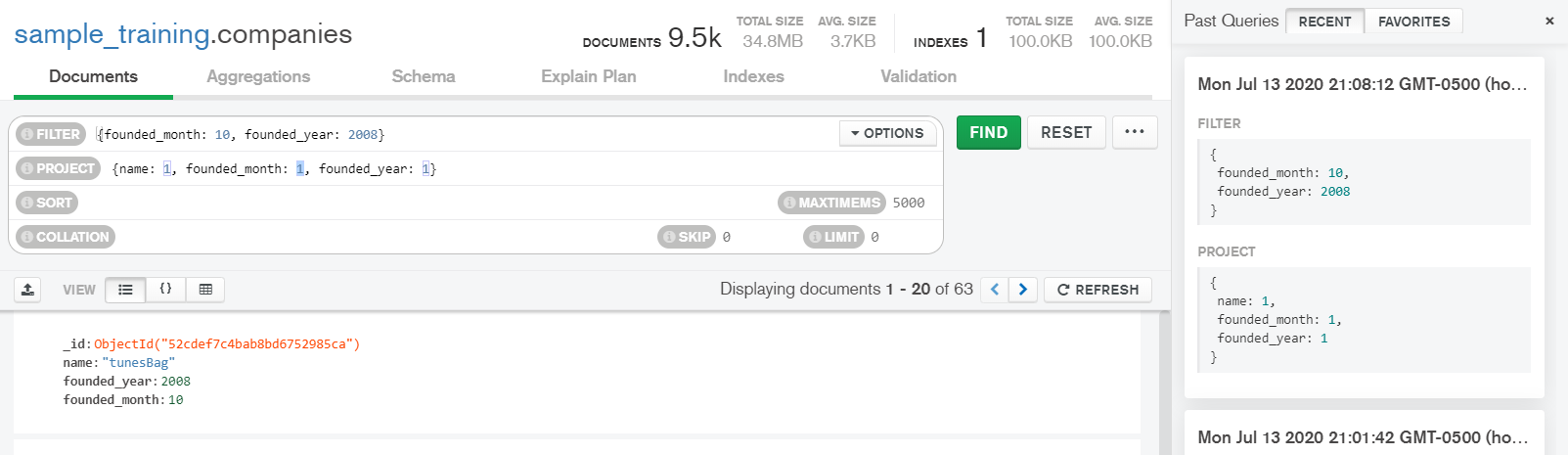
name: 1,

founded\_month: 1,

founded\_year: 1

}

}



* Obtén todas las compañias con más de 50 empleados.

{

filter: {

number\_of\_employees: {

$gt: 50

}

},

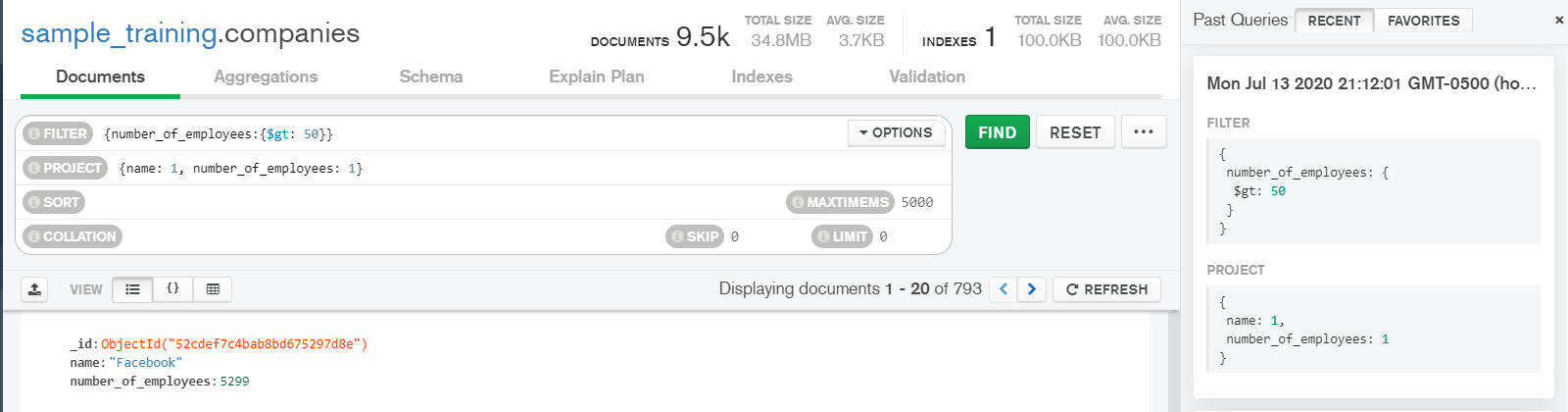
project: {

name: 1,

number\_of\_employees: 1

}

}



* Obtén las historias con número de comentarios entre 10 y 30.

{

filter: {

$and: [

{

comments: {

$gte: 10

}

},

{

comments: {

$lte: 30

}

}

]

},

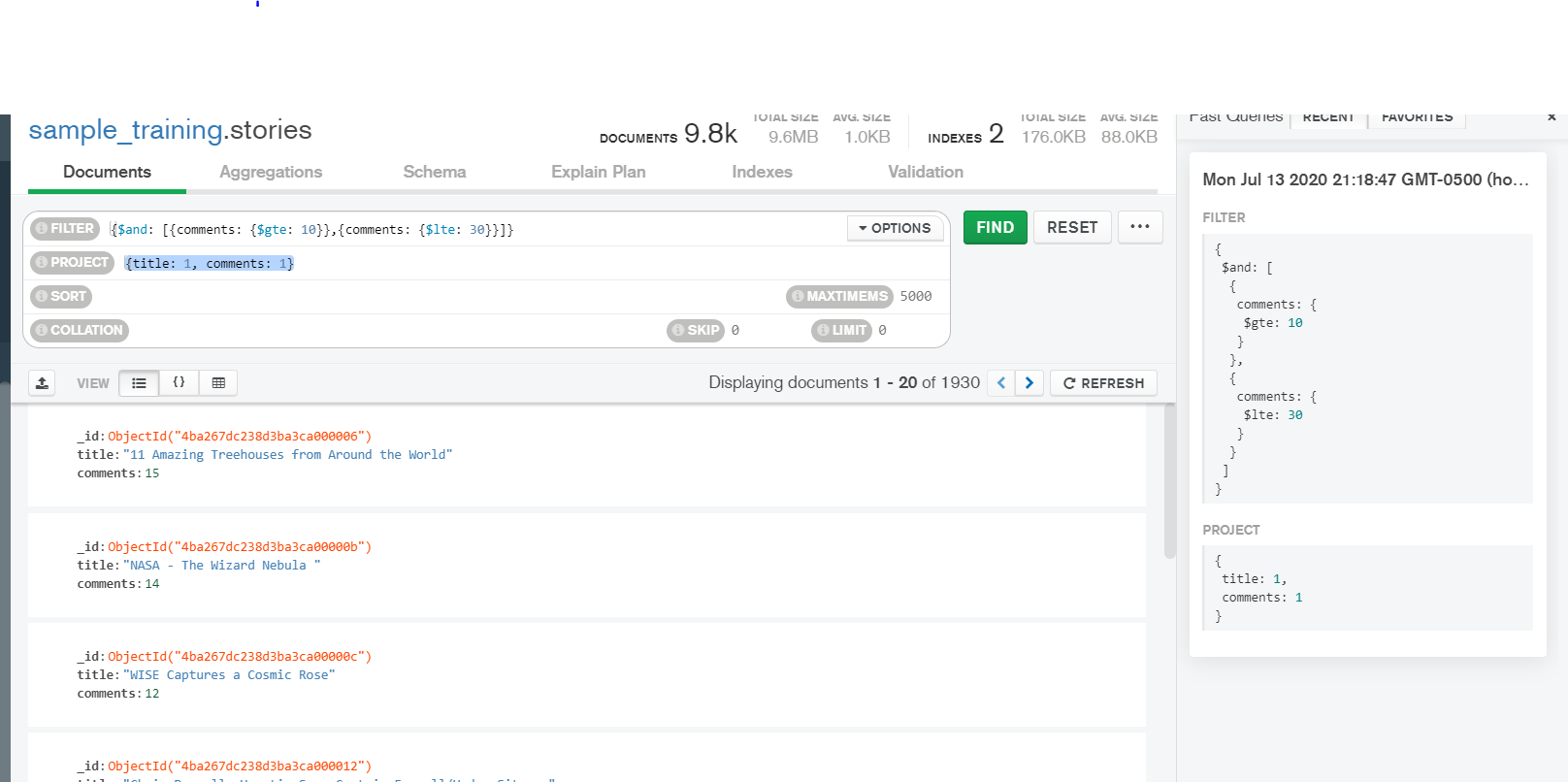
project: {

title: 1,

comments: 1

}

}



* Obtén la empresa con el menor número de empleados.

{

filter: {

$and: [

{

number\_of\_employees: {

$ne: null

}

},

{

number\_of\_employees: {

$ne: 0

}

}

]

},

project: {

name: 1,

number\_of\_employees: 1

},

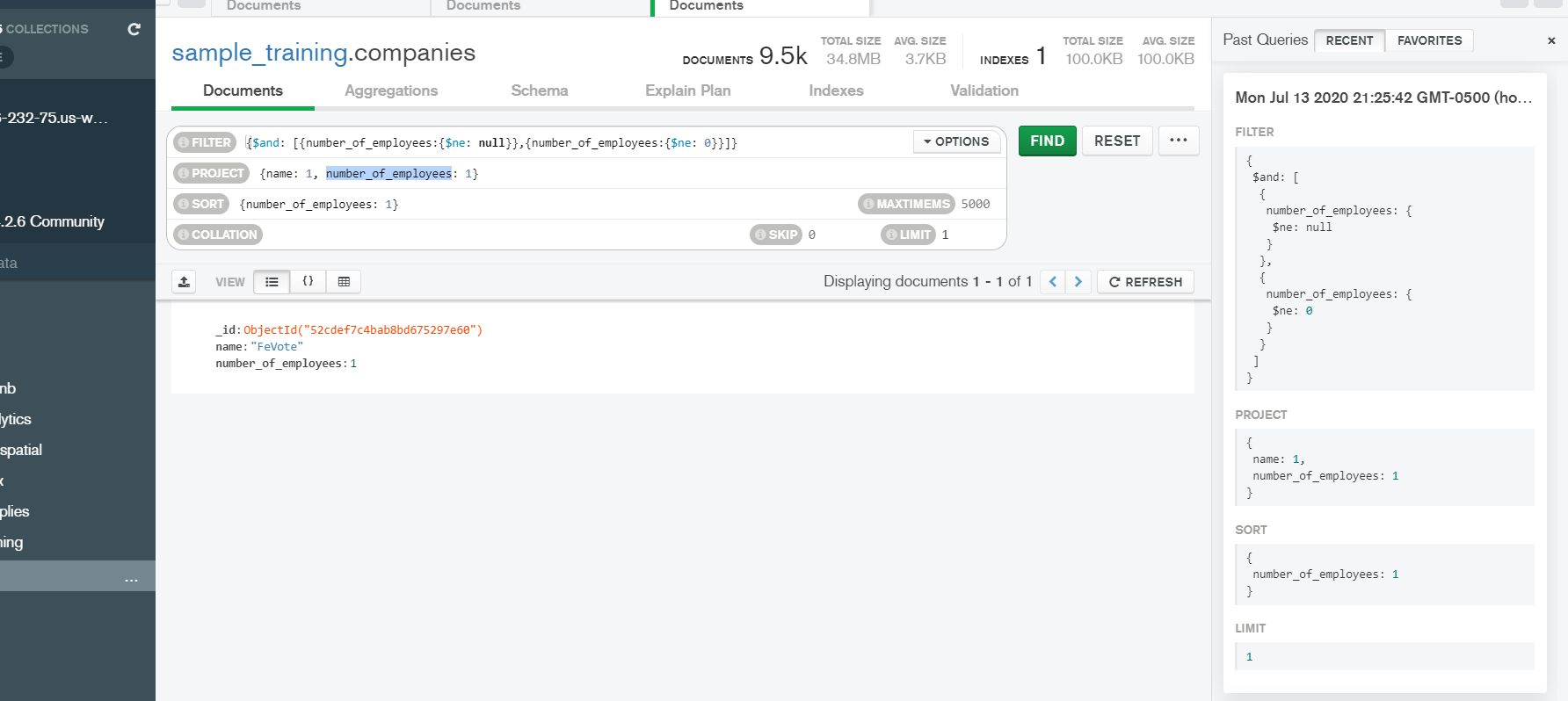
sort: {

number\_of\_employees: 1

},

limit: 1

}



* Obtén la empresa con el mayor número de empleados.

{

filter: {

$and: [

{

number\_of\_employees: {

$ne: null

}

},

{

number\_of\_employees: {

$ne: 0

}

}

]

},

project: {

name: 1,

number\_of\_employees: 1

},

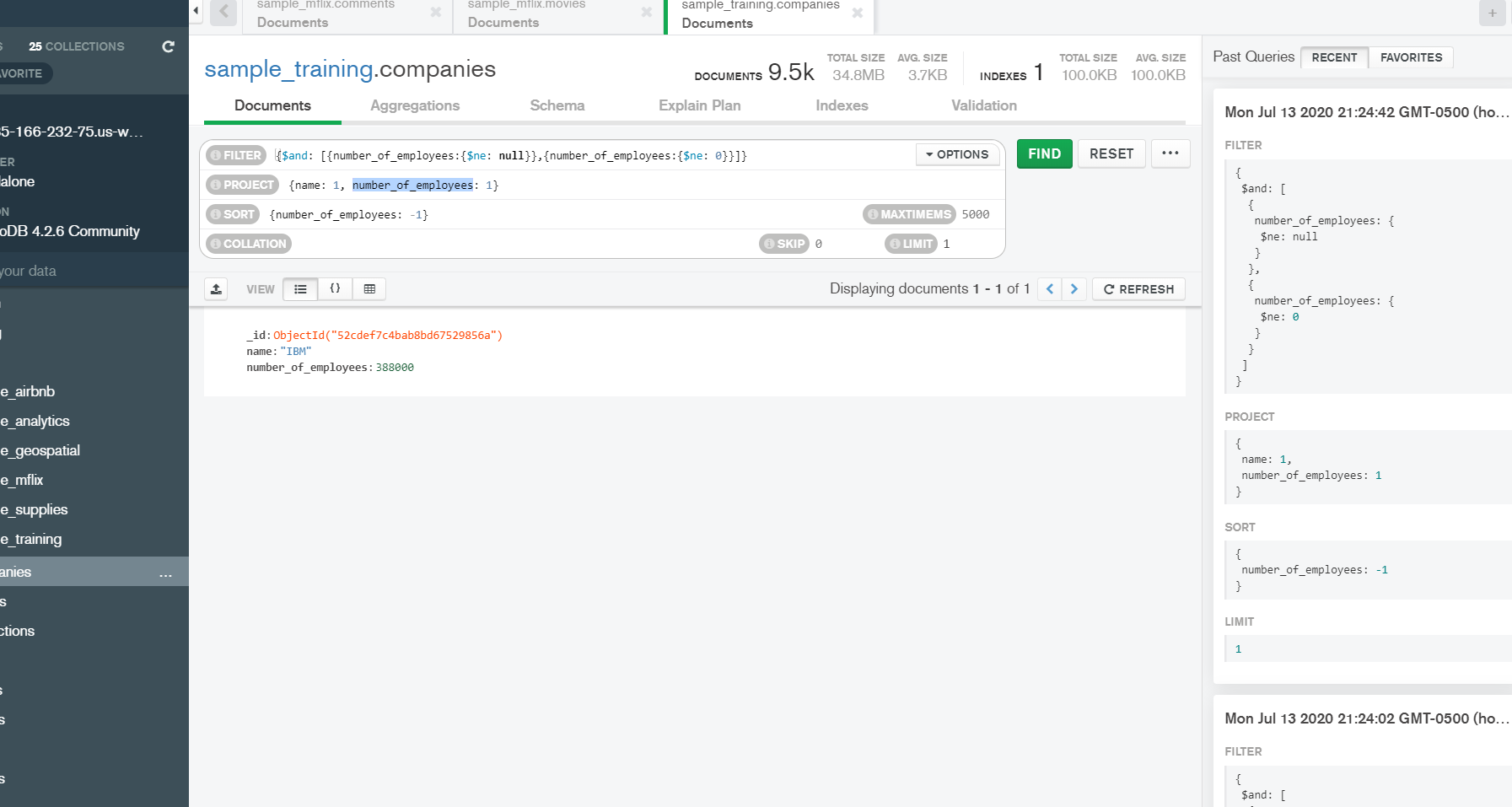
sort: {

number\_of\_employees: -1

},

limit: 1

}



* Obtén la historia más comentada.

{

filter: {

$and: [

{

comments: {

$ne: null

}

},

{

comments: {

$ne: 0

}

}

]

},

project: {

title: 1,

comments: 1

},

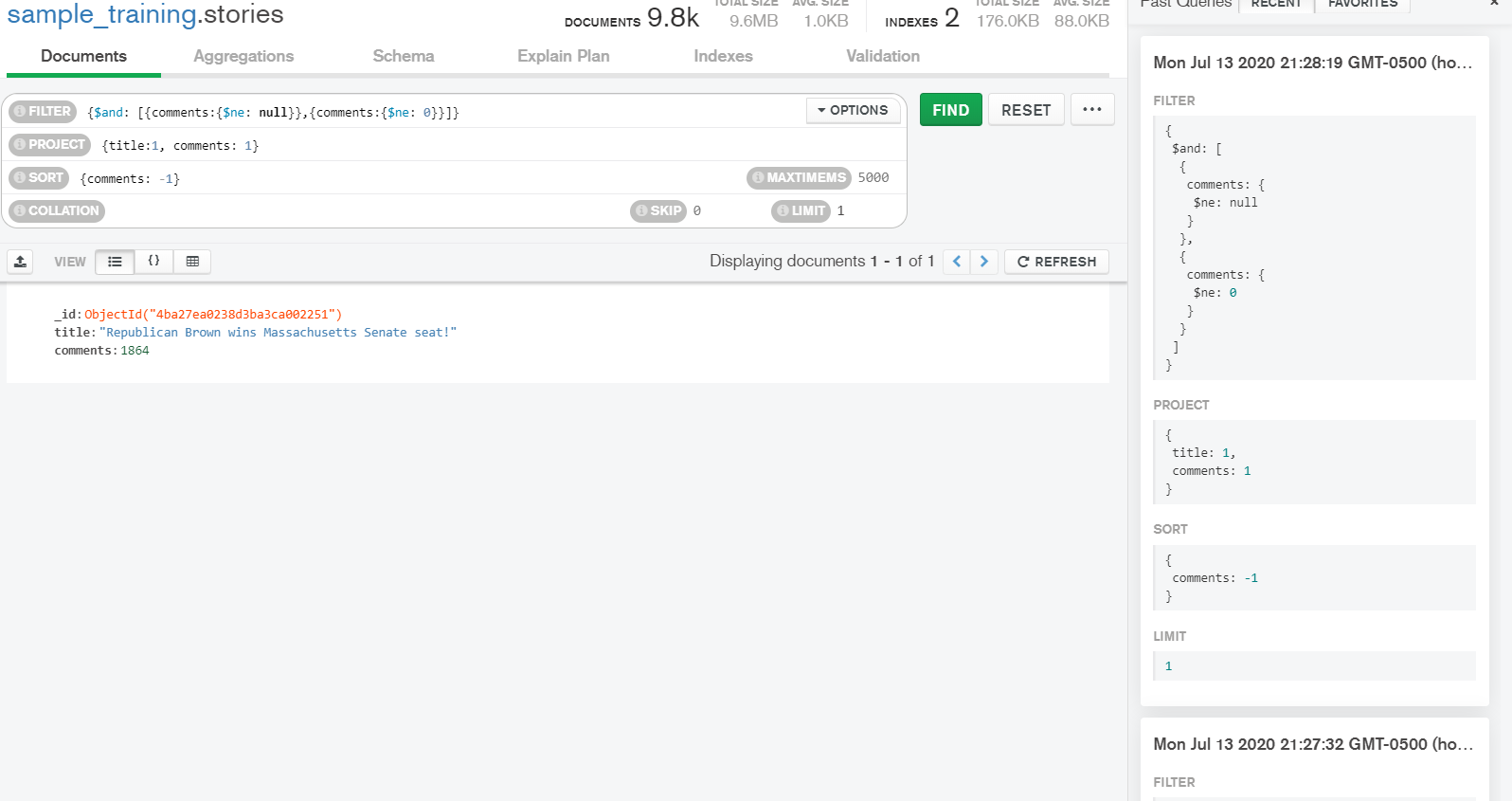
sort: {

comments: -1

},

limit: 1

}



* Obtén la historia menos comentada.

{

filter: {

$and: [

{

comments: {

$ne: null

}

},

{

comments: {

$ne: 0

}

}

]

},

project: {

title: 1,

comments: 1

},

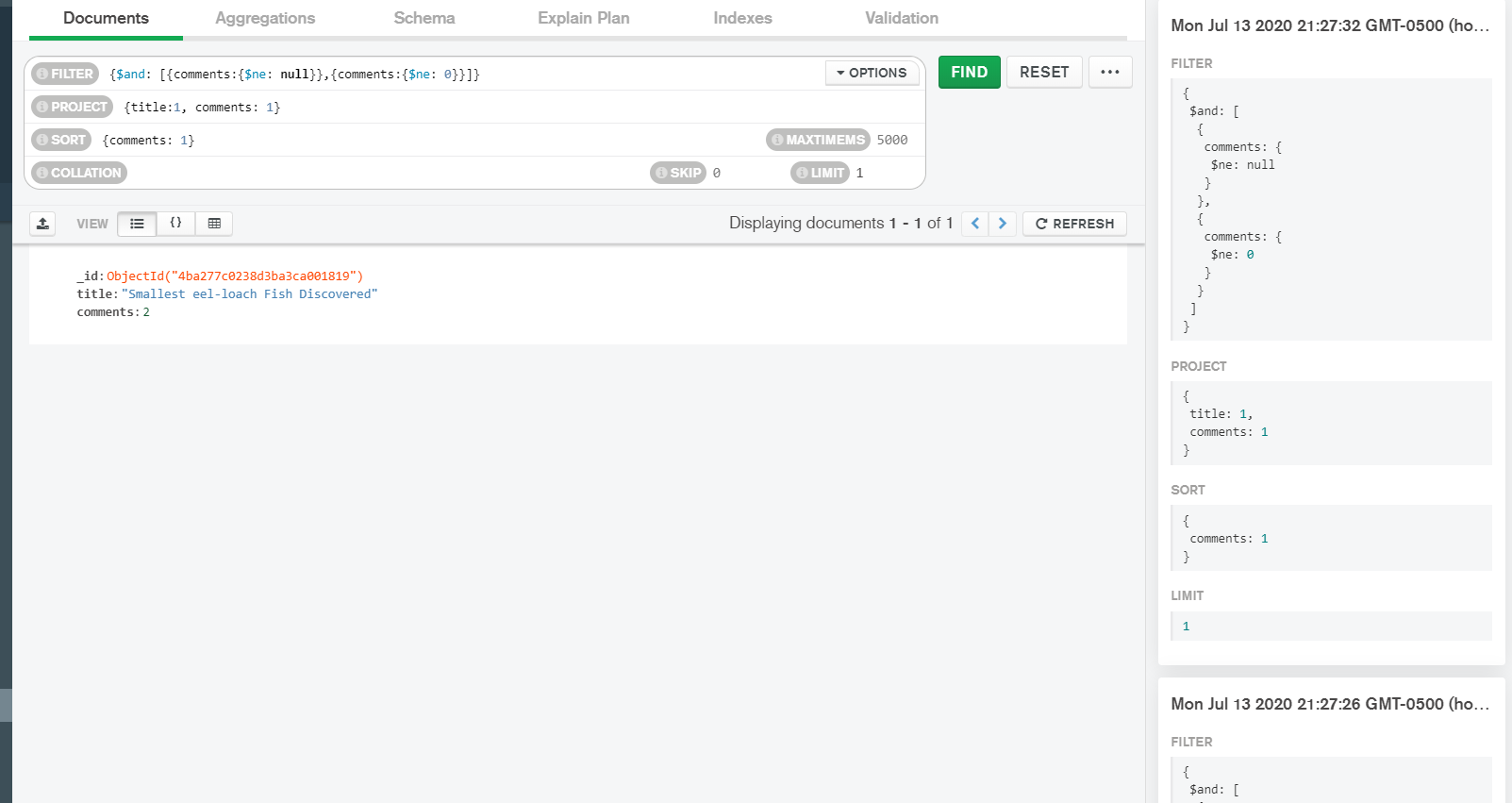
sort: {

comments: 1

},

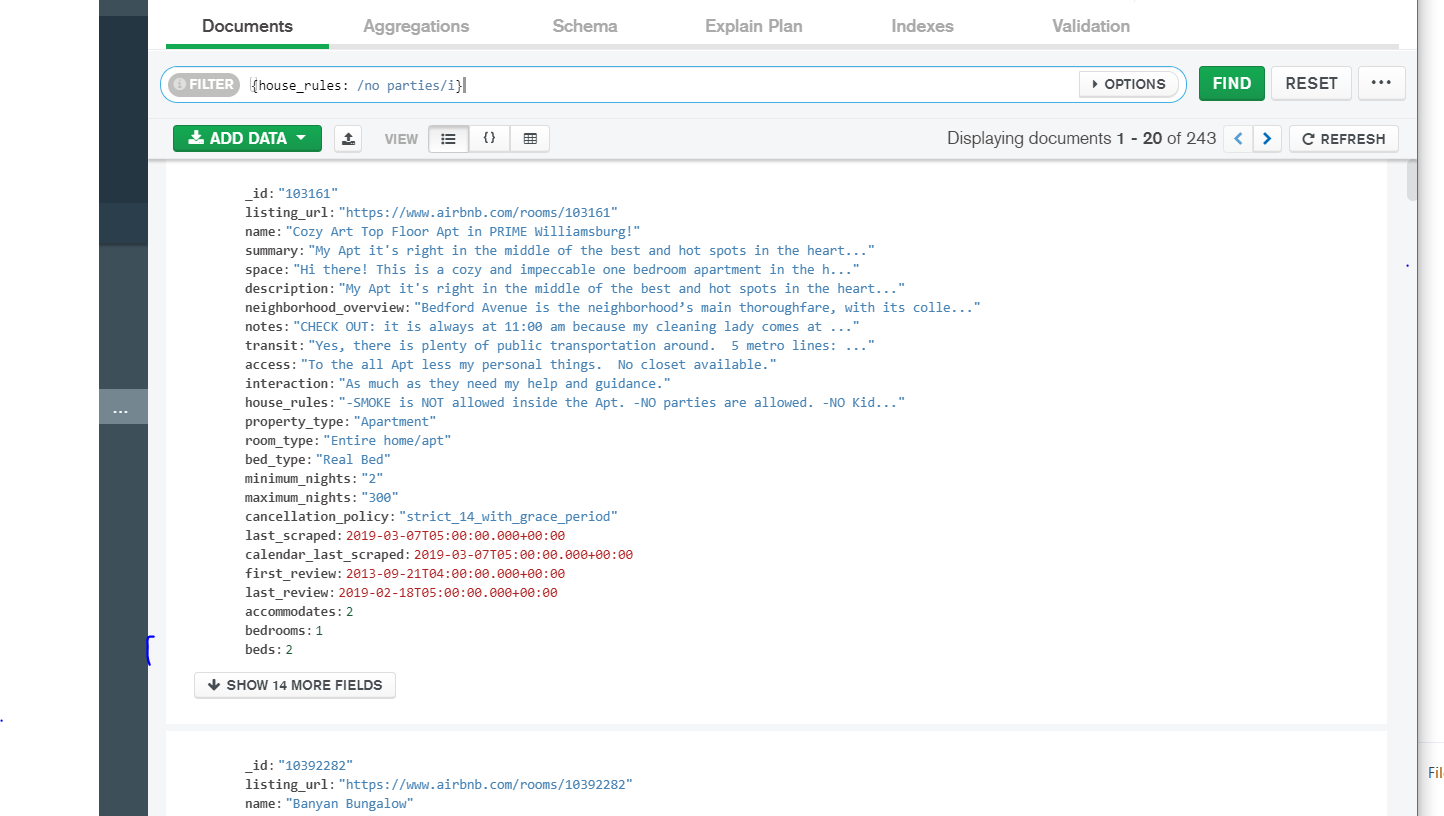
limit: 1

}

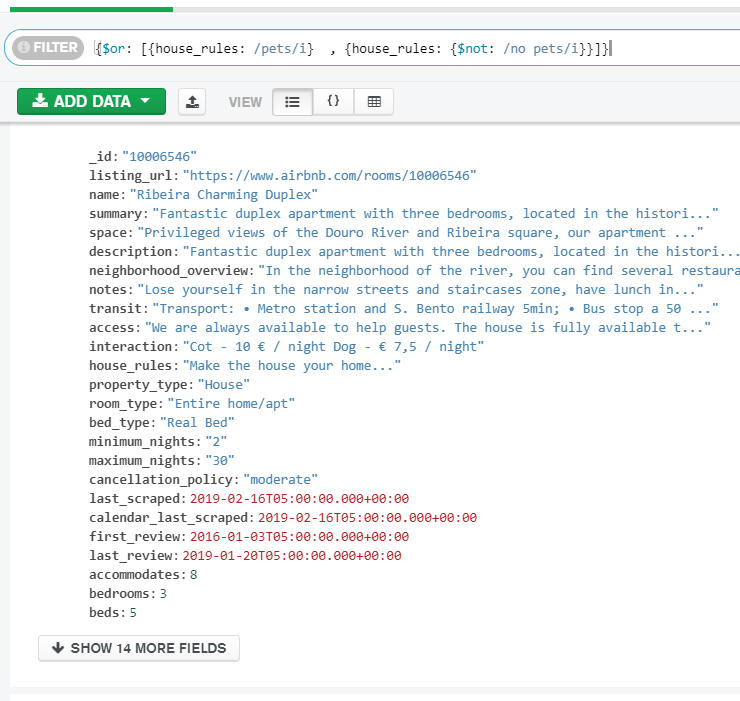


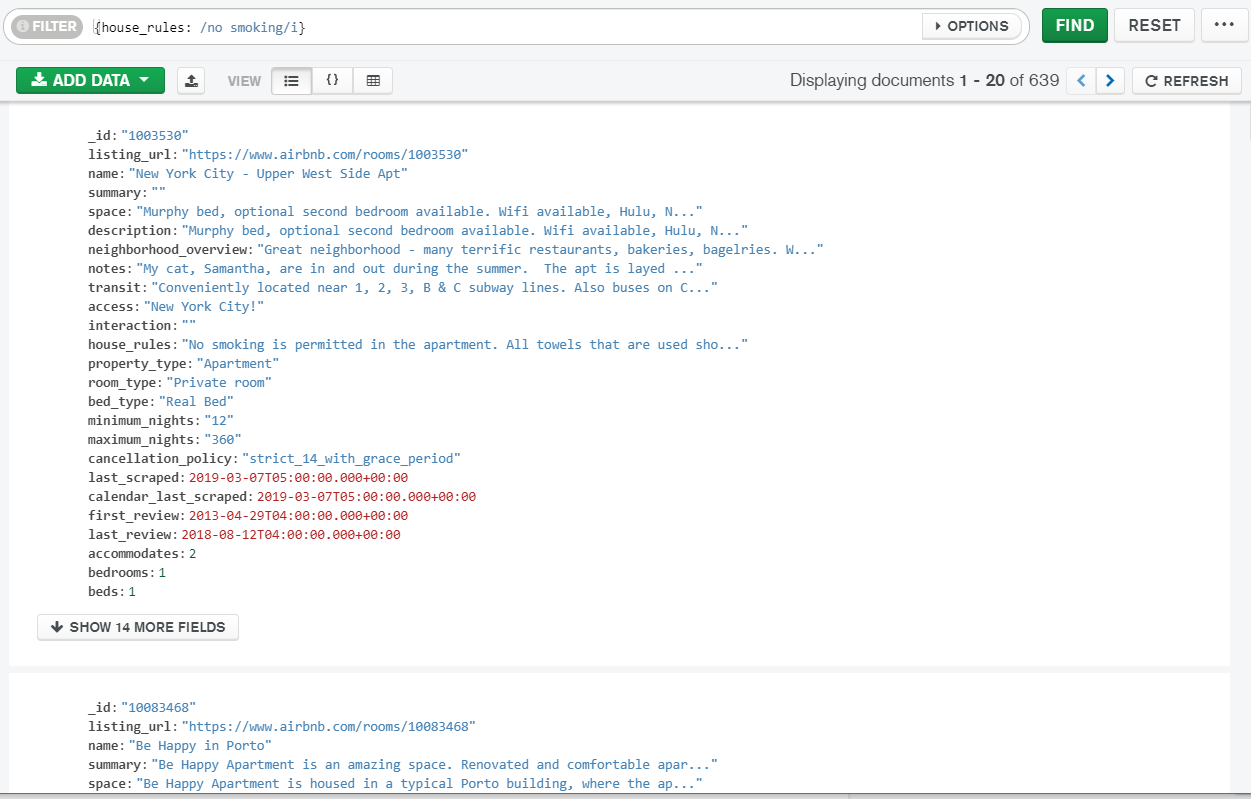
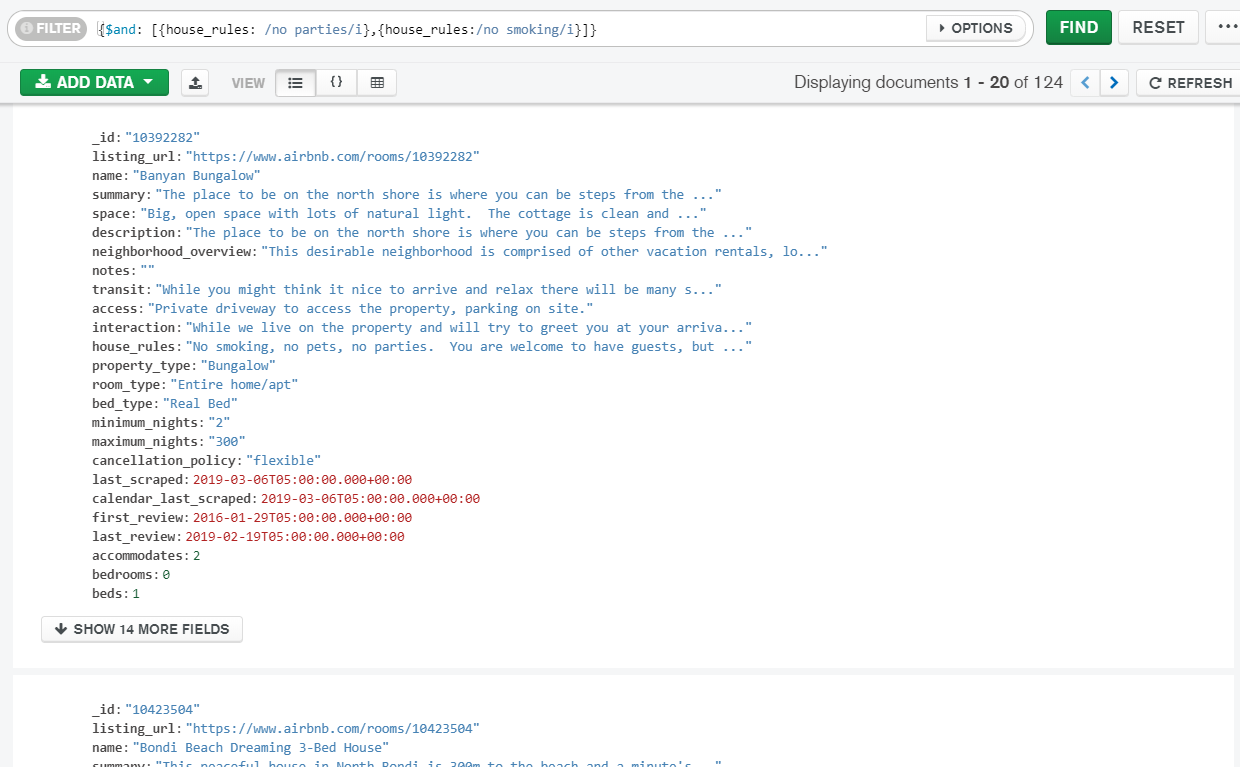
SESION 4

* Propiedades que no permitan fiestas.



* Propiedades que admitan mascotas.

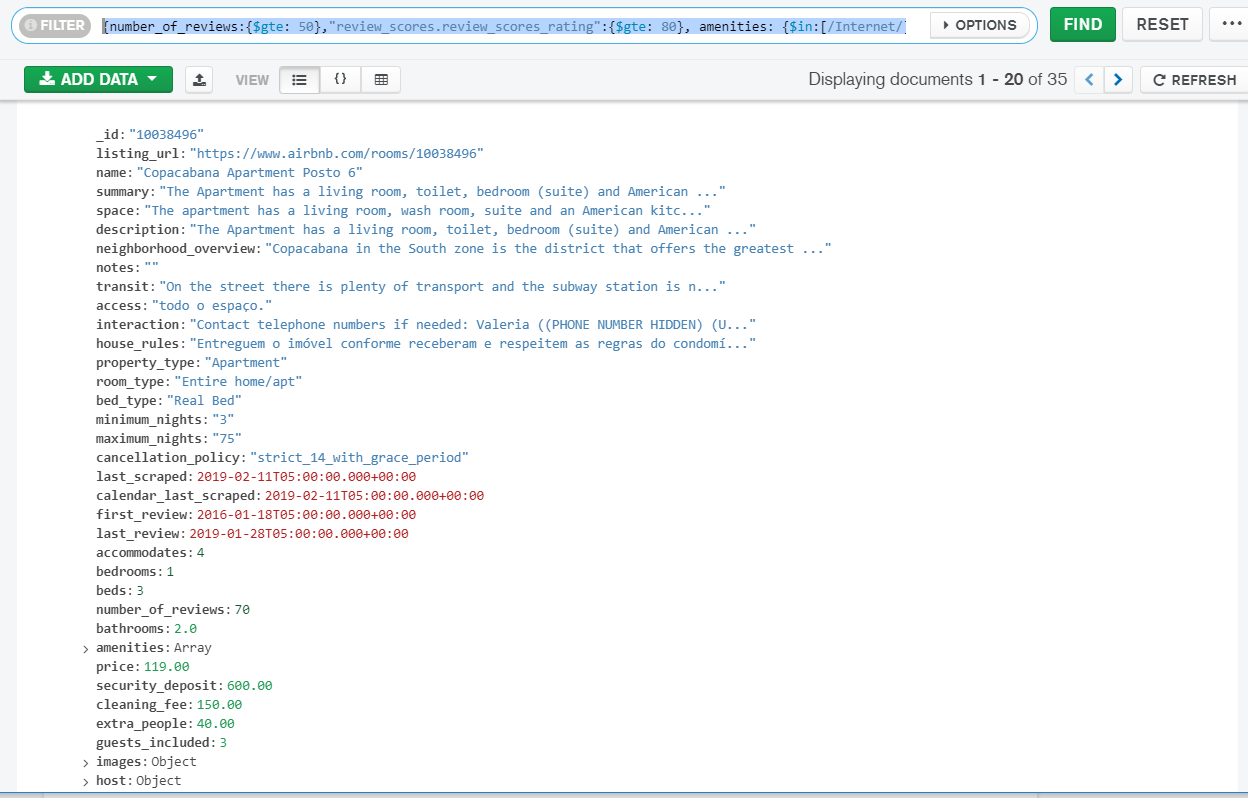


* Propiedades que no permitan fumadores.
* 
* Propiedades que no permitan fiestas ni fumadores.
* 

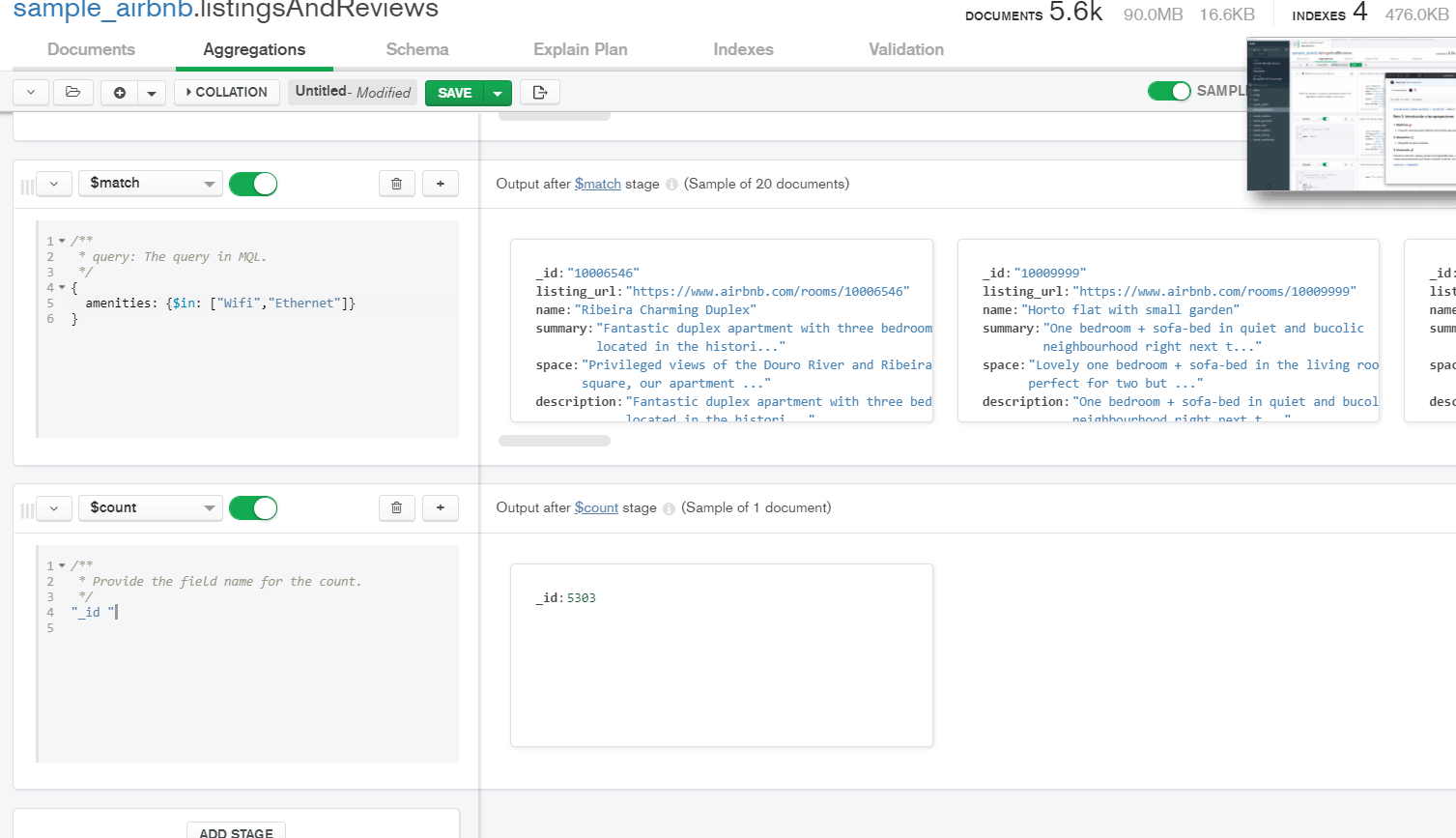
{ price: { $lte: 100 }, "address.country": "Spain", "review\_scores.review\_scores\_rating": { $gte: 50 }, amenities: { $in: [ "Internet", "Wifi"] }, amenities: { $in: ["Elevator"] } }

Usando la colección sample\_airbnb.listingsAndReviews, agrega un filtro que permita obtener todas las publicaciones que tengan 50 o más comentarios, que la valoración sea mayor o igual a 80, que cuenten con conexión a Internet vía cable y estén ubicada en Brazil.

{number\_of\_reviews:{$gte: 50},"review\_scores.review\_scores\_rating":{$gte: 80}, amenities: {$in:[/Internet/]},"address.country": "Brazil"}



Usando la colección sample\_airbnb.listingsAndReviews, mediante el uso de agregaciones, encontrar el número de publicaciones que tienen conexión a Internet, sea desde Wifi o desde cable (Ethernet)



PROYECTO 4

Para este proyecto deberás practicar en el uso de agregaciones, pues serán usadas durante la siguiente sesión.

La base de datos y colección que debes usar es sample\_airbnb.listingsAndReviews.

El proyecto consiste en obtener todas las publicaciones que tengan 50 o más comentarios, que la valoración sea mayor o igual a 80, que cuenten con conexión a Internet vía cable y estén ubicadas en Brazil.

[{

$match: {

number\_of\_reviews: {

$gte: 50

}

}

}, {

$match: {

"review\_scores.review\_scores\_rating": {

$gte: 80

}

}

}, {

$match: {

amenities: {

$in: [/Internet/]

}

}

}, {

$match: {

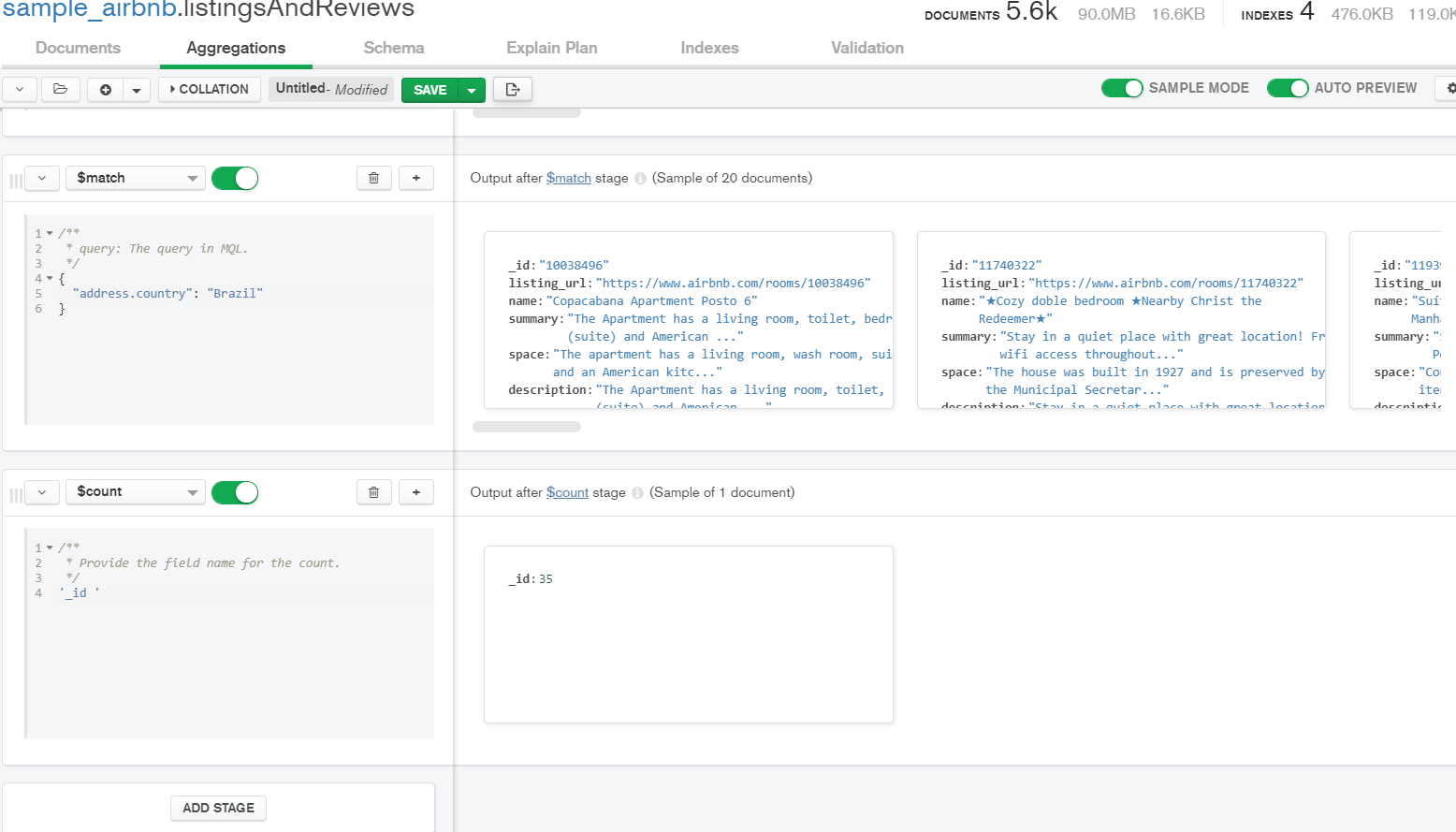
"address.country": "Brazil"

}

}, {

$count: '\_id '

}]



SESION 5

Reto 1

Con base en el ejemplo 1, modifica el agrupamiento para que muestre el costo promedio por habitación por país de las propiedades de tipo casa.

[{

$match: {

property\_type: "House",

bedrooms: {

$gte: 1

}

}

}, {

$addFields: {

costo\_recamara: {

$divide: ['$price', '$bedrooms']

}

}

}, {

$group: {

\_id: "$address.country",

recamaras: {

$sum: 1

},

total: {

$sum: "$costo\_recamara"

}

}

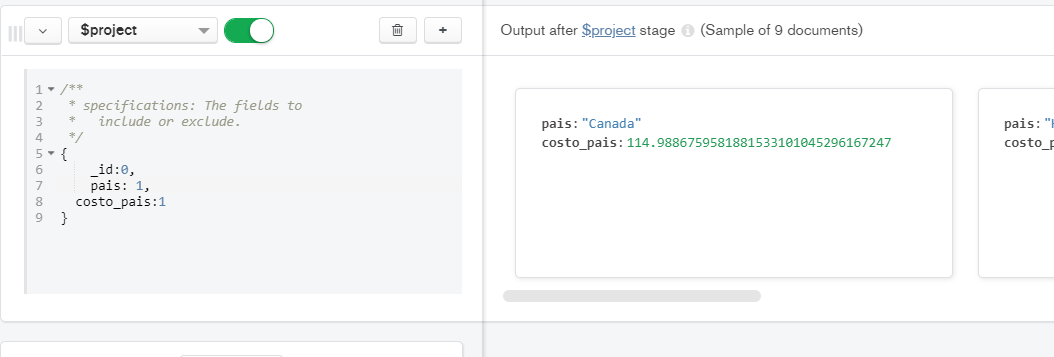
}, {

$addFields: {

pais: "$\_id",

costo\_pais: {

$divide: ['$total', '$recamaras']



Reto 2

[{

$lookup: {

from: 'users',

localField: 'email',

foreignField: 'email',

as: 'usuario'

}

}, {

$addFields: {

usuario\_objeto: {

$arrayElemAt: ["$usuario", 0]

}

}

}, {

$addFields: {

usuario\_password: "$usuario\_objeto.password"

}

}, {

$project: {

\_id: 0,

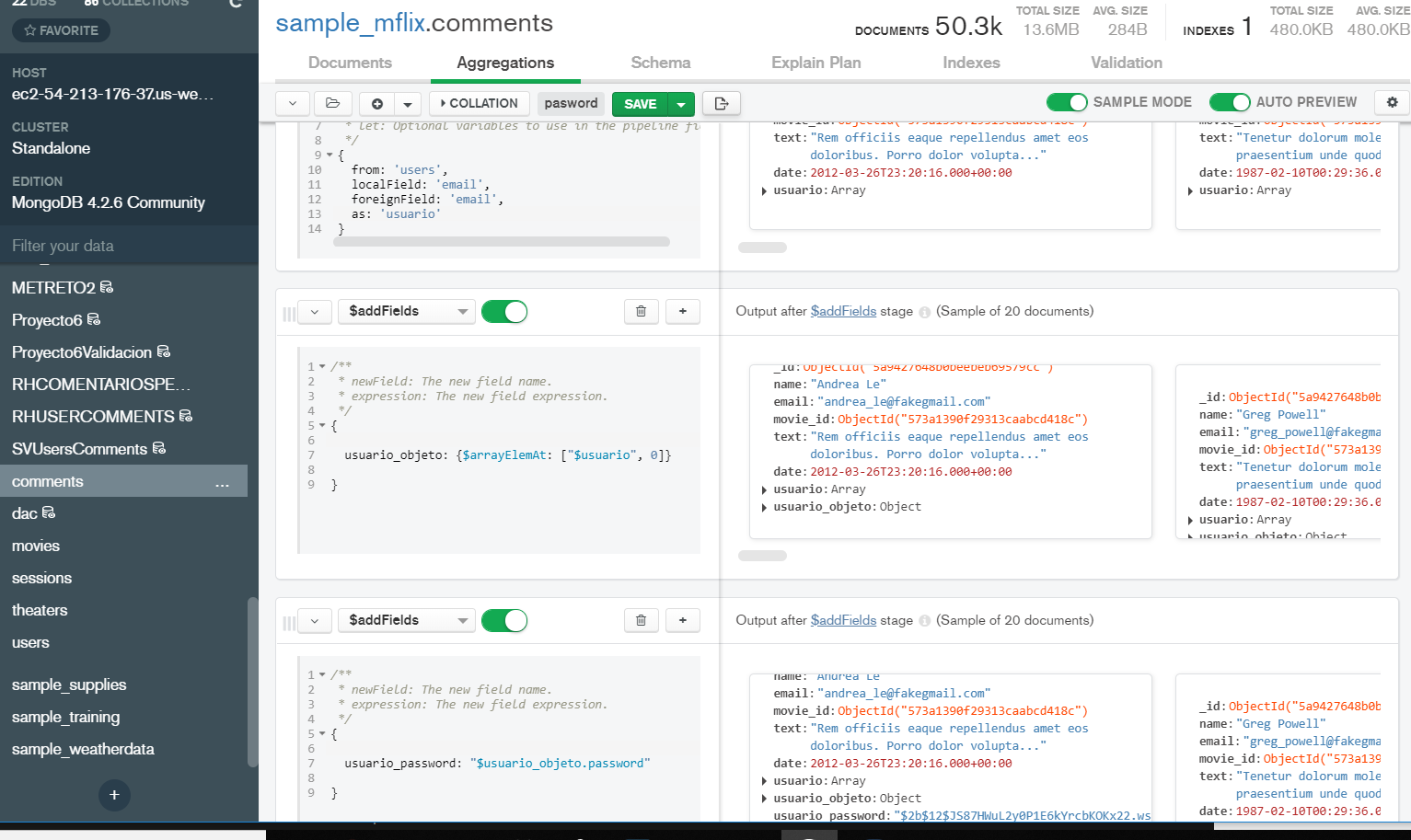
name: 1,

email: 1,

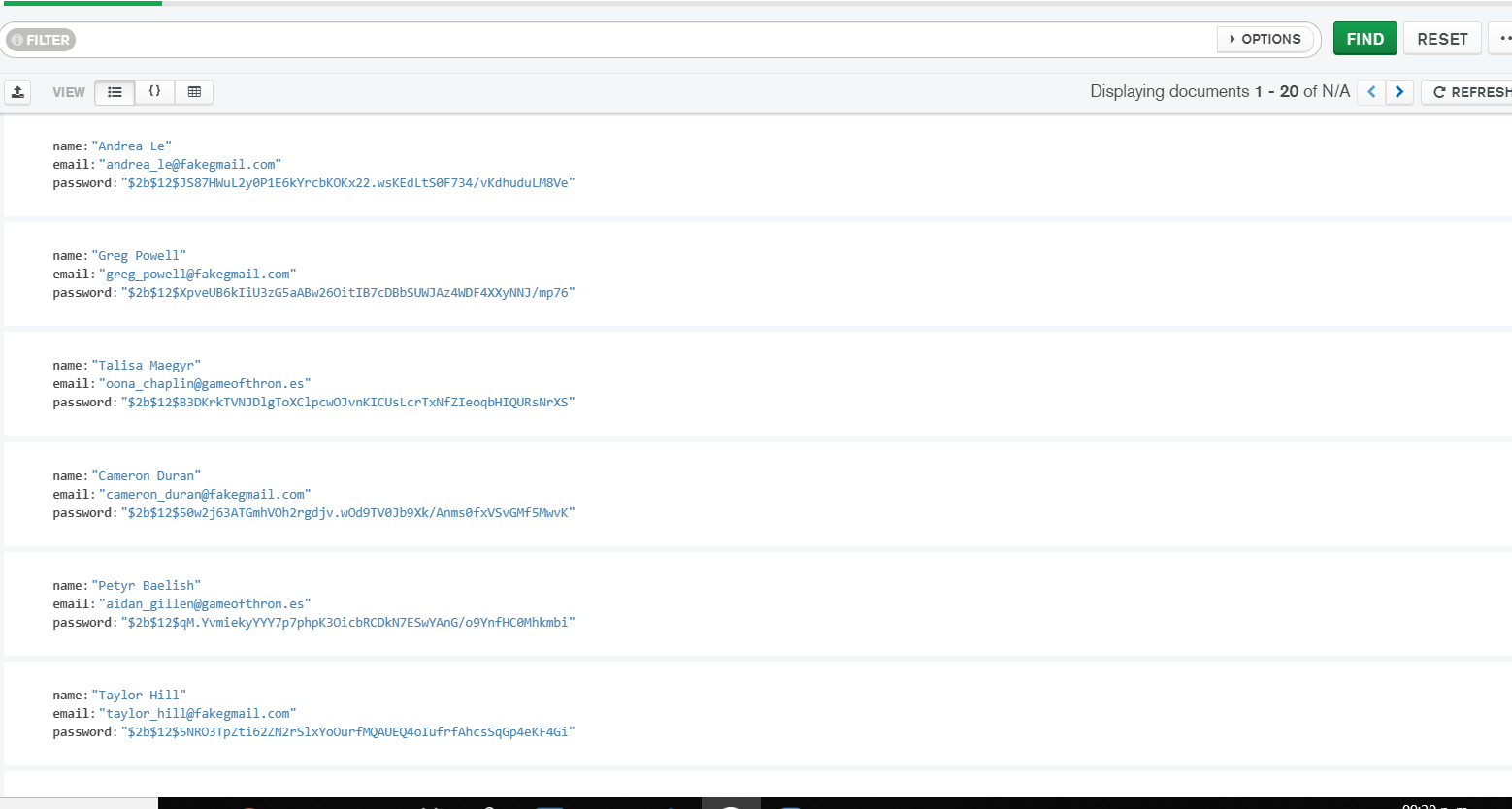
usuario\_password: 1

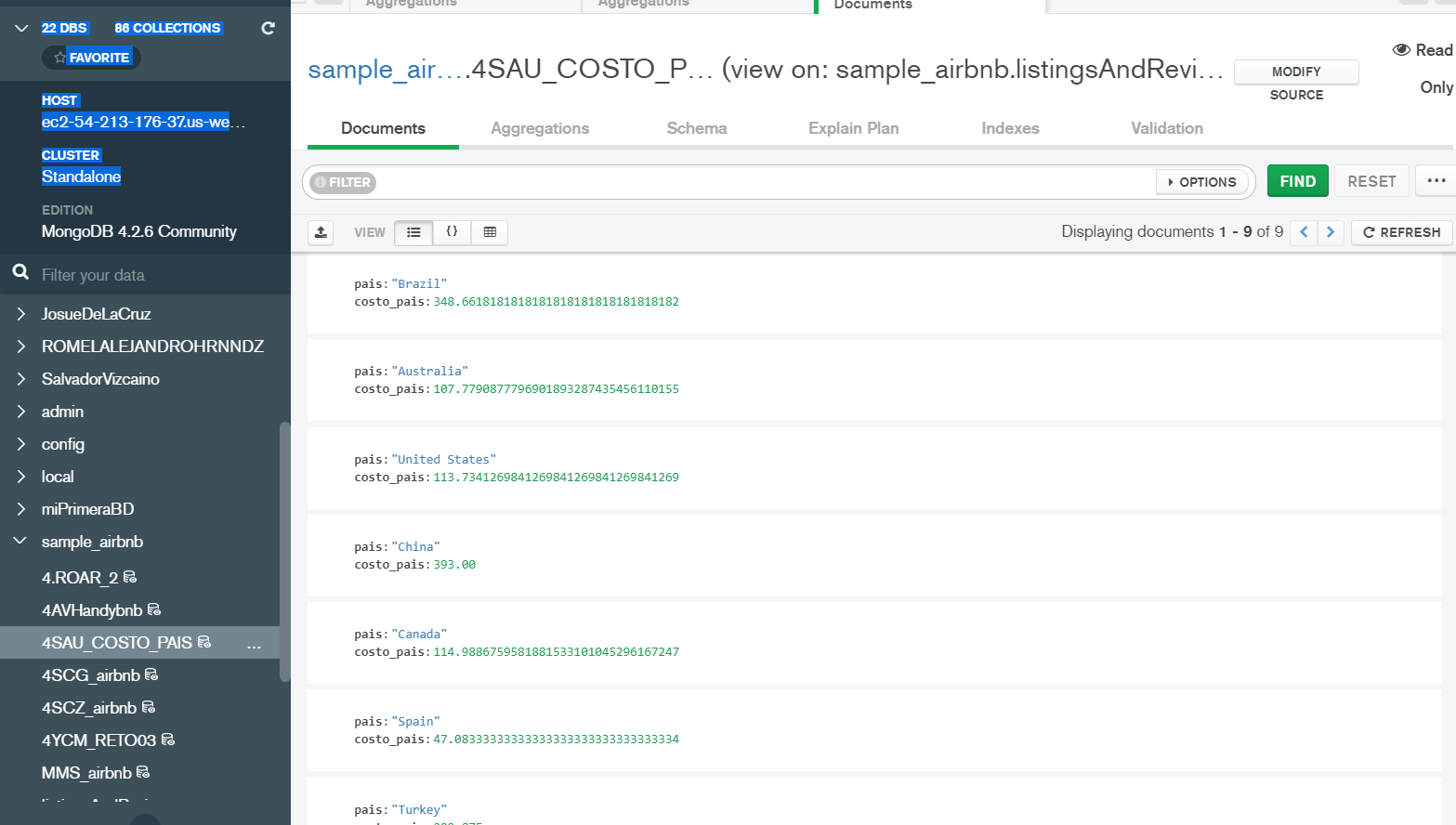
}

}]



Reto 3





PROYECTO 5

[{

$unwind: {

path: "$genres",

}

}, {

$unwind: {

path: "$countries"

}

}, {

$group: {

\_id: {

age: "$genres",

pais: "$countries"

},

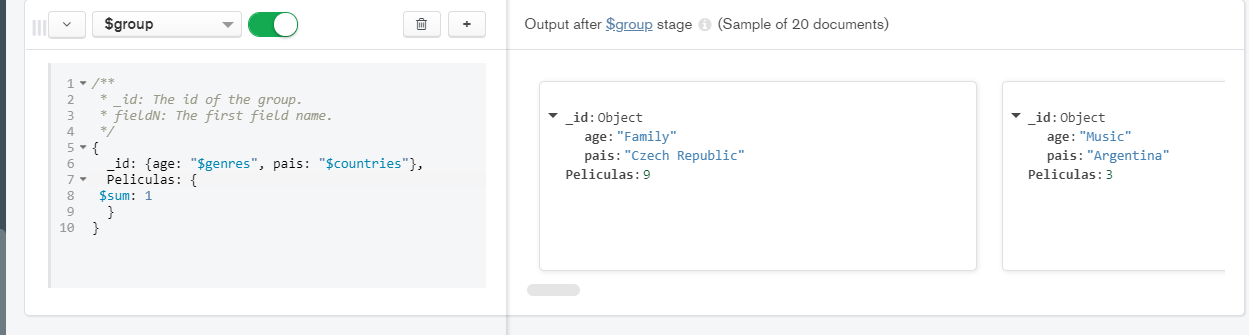
Peliculas: {

$sum: 1

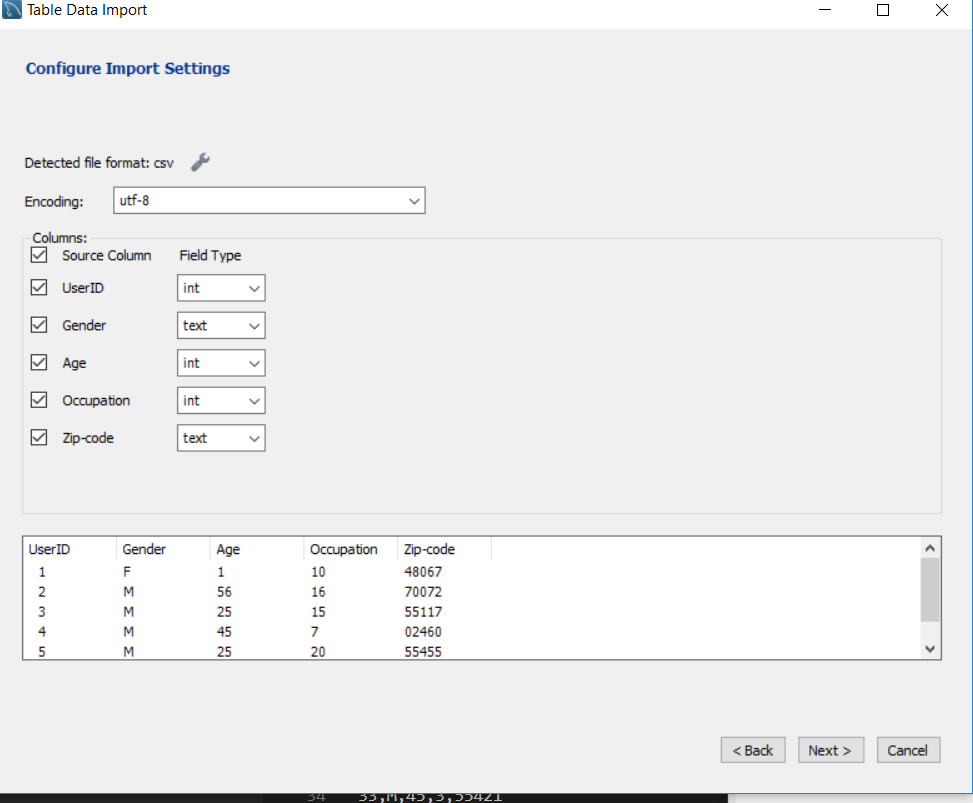
}

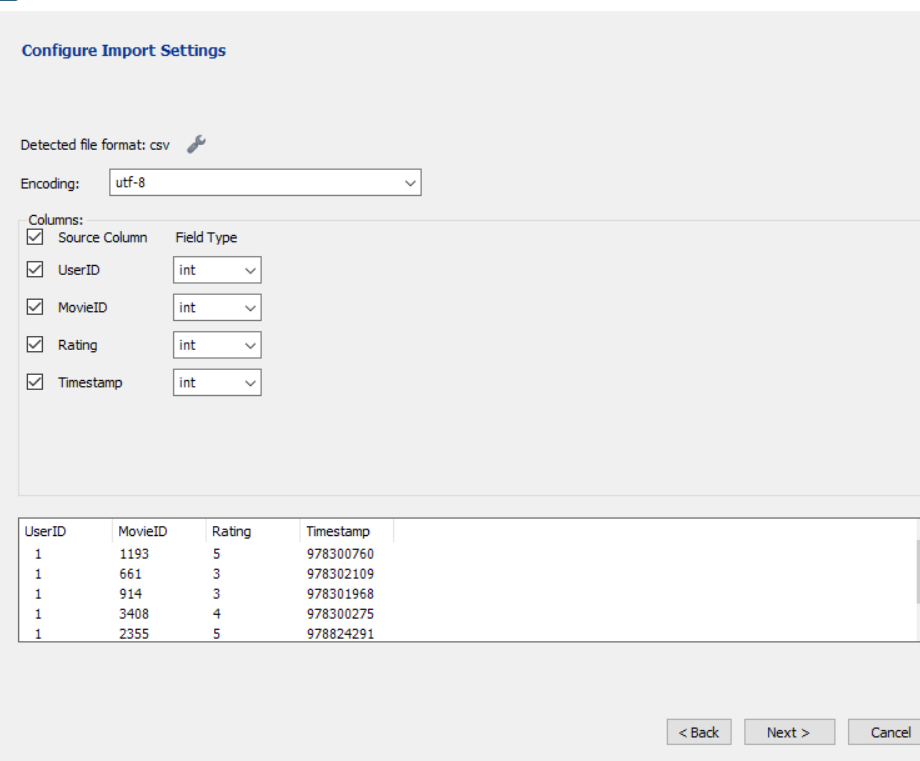
}

}]

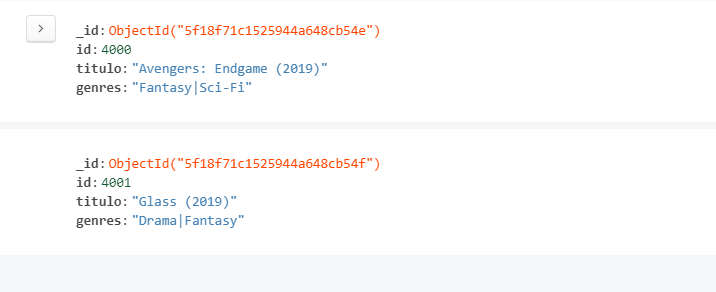


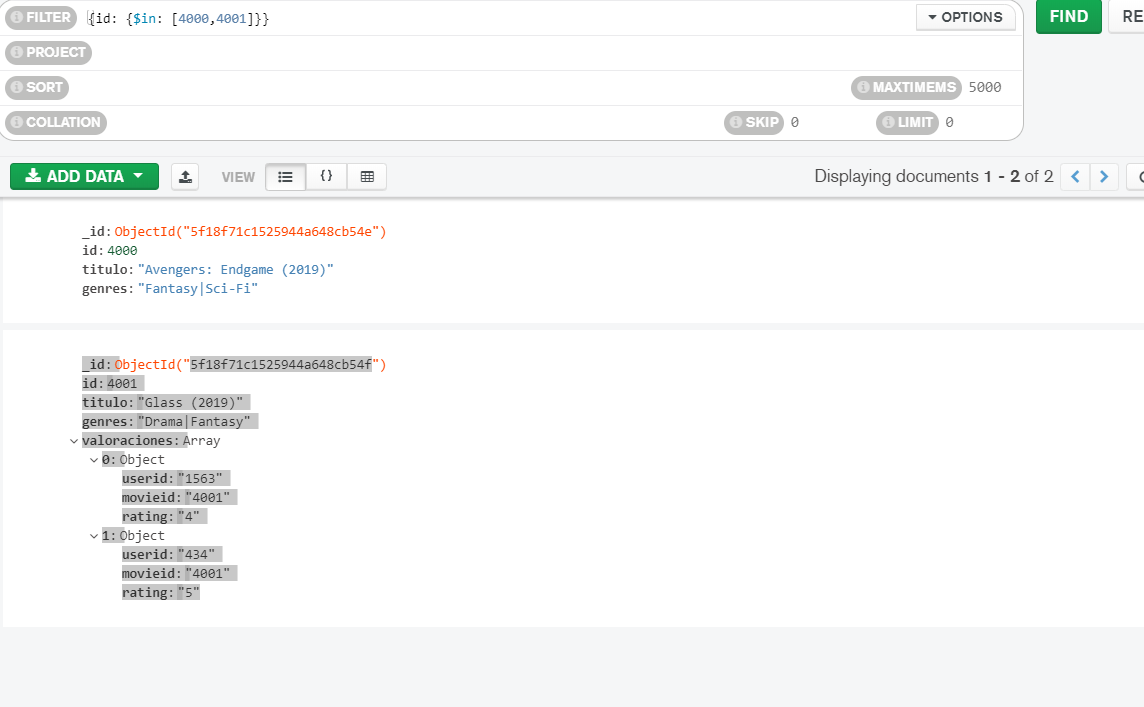
SESION 6





PROYECTO 6





SESION 8

Reto 1

[{$match: {

$and: [

{Latitude:{$gte: 19.5}},

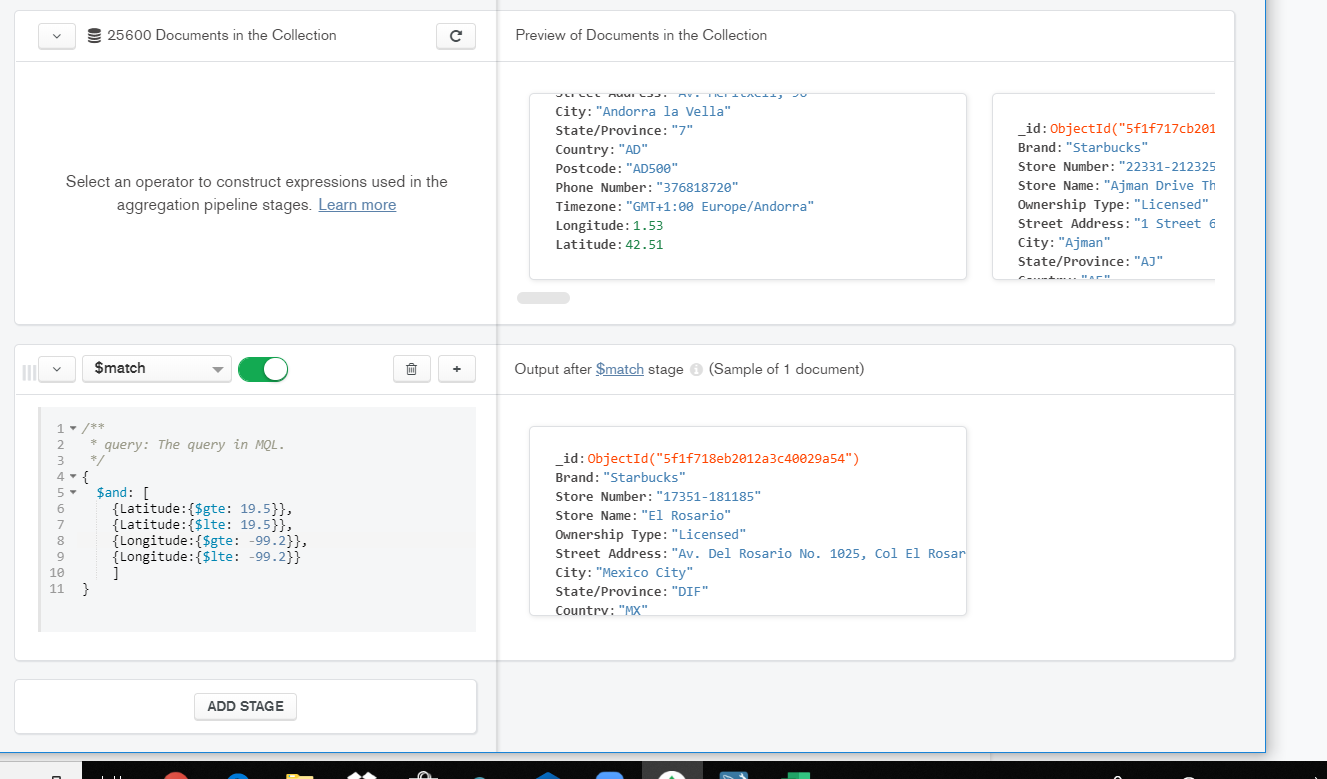
{Latitude:{$lte: 19.5}},

{Longitude:{$gte: -99.2}},

{Longitude:{$lte: -99.2}}

]

}}]



Reto 2

* ¿Cuál fue el país con mayor número de muertes?

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

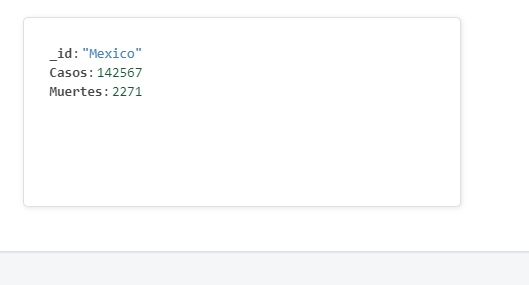
}}, {$sort: {

Muertes: -1

}}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$limit: 1}]



* ¿Cuál fue el país con menor número de muertes?

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

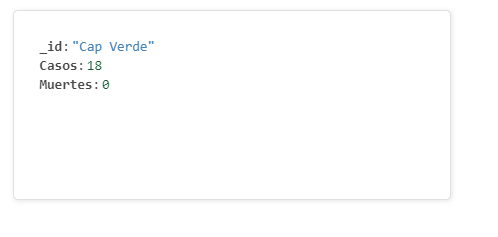
}}, {$sort: {

Muertes: 1

}}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$limit: 1}]



* ¿Cuál fue el país con el mayor número de casos?

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

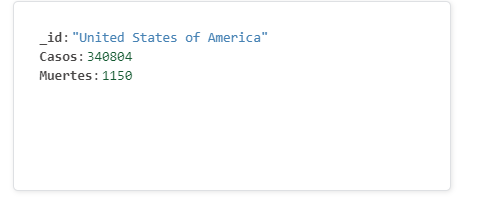
}}, {$sort: {

Casos: -1

}}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$limit: 1}]



* ¿Cuál fue el país con el menor número de casos?

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

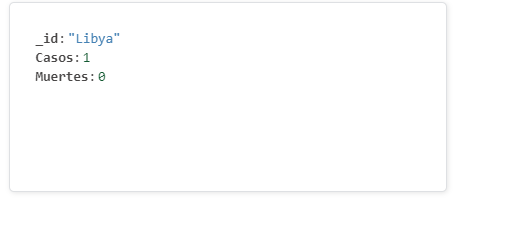
}}, {$sort: {

Casos: 1

}}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$limit: 1}]



* ¿Cuál fue el número de muertes promedio?

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

}

}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$group: {

\_id: null,

Casos: {

$sum: '$Casos'

},

Muertes: {$sum: '$Muertes'},

Paises: {$sum: 1}

}}, {$addFields: {

Muertes\_prom: {$divide: ['$Muertes', '$Paises']}

}}, {$project: {

\_id:0,

Muertes\_prom: 1

}}]



* ¿Cuál fue el número de casos promedio?

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

}

}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$group: {

\_id: null,

Casos: {

$sum: '$Casos'

},

Muertes: {$sum: '$Muertes'},

Paises: {$sum: 1}

}}, {$addFields: {

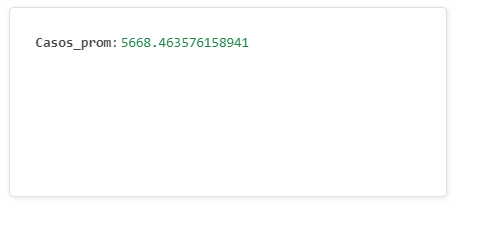
Casos\_prom: {$divide: ['$Casos', '$Paises']}

}}, {$project: {

\_id:0,

Casos\_prom:1

}}]



* Top 5 de países con más muertes

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

}

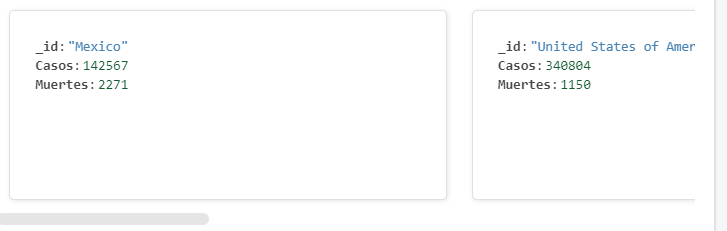
}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$sort: {

Muertes: -1

}}, {$limit: 5}]



* Top 5 de países con menos muertes

[{$group: {

\_id: '$Country',

Casos: {

$sum: '$Cases'

},

Muertes: {$sum: '$Deaths'}

}

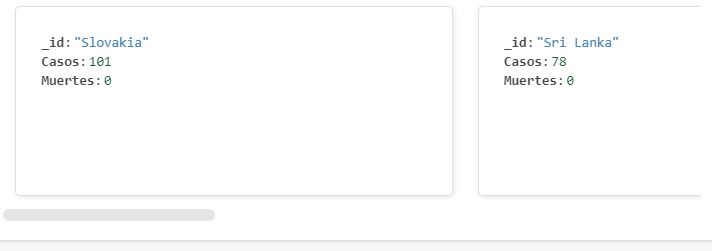
}, {$match: {

\_id: {$nin: [/Grand Total/]}

}}, {$sort: {

Muertes: 1

}}, {$limit: 5}]



RETO 3

* ¿Cuál es país con mayor número de casos?

[{$group: {

\_id: '$Region',

Casos: {

$sum: '$Confirmed'

},

Muertes: {

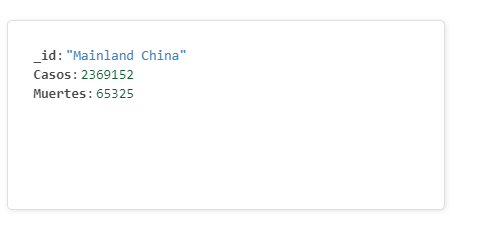
$sum: '$Deaths'

}

}}, {$sort: {

Casos: -1

}}, {$limit: 1}]



* ¿Cuál es el país con mayor número de muertes?

[{$group: {

\_id: '$Region',

Casos: {

$sum: '$Confirmed'

},

Muertes: {

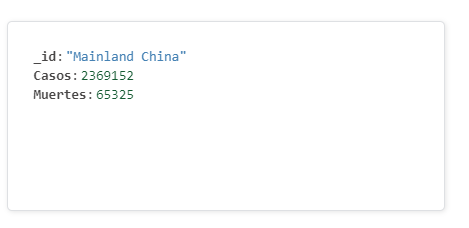
$sum: '$Deaths'

}

}}, {$sort: {

Muertes: -1

}}, {$limit: 1}]



* Usando las coordenadas, encuentra el epicentro del virus.

[{$group: {

\_id: '$Province',

Lat: {

$max: '$Lat'

},

Long: {

$max: '$Long'

}

}}, {$group: {

\_id: null,

TLat: {

$sum: '$Lat'

},

TLong: {

$sum: '$Long'

}

,

TP: {

$sum: 1

}

}

}, {$addFields: {

Prom\_lat: {$divide: ['$TLat','$TP']},

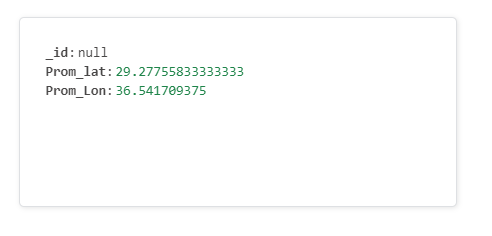
Prom\_Lon: {$divide: ['$TLong','$TP']}

}}, {$project: {

Prom\_lat: 1,

Prom\_Lon:1

}}]



* Usando el epicentro, encuentra las 5 regiones más cercanas a dicho epicentro

[{$match: {

Lat: {$ne: ""},

Long: {$ne: ""}

}}, {$group: {

\_id: '$Region',

Latitude: {

$max: '$Lat'

},

Longitude: {$max: '$Long'}

}}, {$addFields: {

Lat: {$convert: {input: '$Latitude', to: 'double'}},

Long: {$convert: {input: '$Longitude', to: 'double'}}

}}, {$match: {

$and: [{Lat: {$gte: 20}}, {Lat: {$lte: 40}},

{Long: {$gte: 25}}, {Long: {$lte: 45}}]

}}, {$limit: 5}]

