

Sesión 1

Clase 1/07/2020

Bases de datos Relacionales.-

Relaciona tablas que contienen datos por medio de un campo llave, con esto se evita el tener que repetir información.

Para la conexión de Mysql usar

ec2-35-166-232-75.us-west-2.compute.amazonaws.com

BEdu_0583

correr al final de la linea del query para evitar ejecutar cosas que no van

Comando para visualizar las bases de datos: show databases;

nota: atajo para ejecutar la consulta ctr+enter

Reto 2

-- cuál es el nombre de los empleados con el puesto 4

*select * from empleado where id_puesto = 4;*

-- qué puestos tienen un salario mayor a 10000

*select * from puesto where salario > 10000;*

-- qué artículos tienen un precio mayor a 1000 y un iva mayor a 100?

*select * from articulo where precio > 1000 and iva > 100;*

-- qué ventas incluyen los artículos 135 o 963 y fueron hechas por los empleados 835 o 369?

*select * from venta where (id_articulo = 135 or id_articulo = 963) and (id_empleado = 835 or id_empleado = 369);*

a diferencia de t-sql para seleccionar un top x, se utiliza limit en vez de top

el limit después de la tabla

el limit se escribe al final de la sintaxis.

-- usando la base de datos tienda, escribe una consulta que permita obtener el top 5 puestos por salario

*select * from puesto order by salario desc limit 5 ;*

Capturas clase 1 (01-07-2020)

```

6 • describe empleado;
7 • describe venta;
8 • describe puesto;
9 • describe articulo;

```

```

8 • describe puesto;
9 • describe articulo;

```

Field	Type	Null	Key	Default	Extra
id_venta	int	NO	PRI	NULL	
id_articulo	int	NO	MUL	NULL	
id_empleado	int	NO	MUL	NULL	
clave	varchar(45)	NO		NULL	
fecha	timestampt	NO		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TI...

Field	Type	Null	Key	Default	Extra
id_puesto	int	NO	PRI	NULL	
nombre	varchar(45)	NO		NULL	
salario	double	NO		NULL	

```

9 • describe articulo;

```

Field	Type	Null	Key	Default	Extra
id_articulo	int	NO	PRI	NULL	
nombre	varchar(45)	NO		NULL	
precio	double	NO		NULL	
iva	double	NO		NULL	
cantidad	int	NO		0	

```

25 -- Reto 2 Estructura basica de una consulta
26 • select * from empleado where id_puesto = 4;
27
28 • select * from puesto where salario > 10000;

```

id_empleado	id_puesto	nombre	apellido_paterno	apellido_materno	rfc
410	100	Honor	Parsonage	Lowdyane	YQVPO07255G64
307	200	Suzy	Ingerfield	Aire	SJUT182565W08
310	200	Harvey	Deeney	Kennerley	2AHQ035377H15
378	200	Damita	Fishburn	Rait	WQZR682964E50
NULL	NULL	NULL	NULL	NULL	NULL

```

28 • select * from puesto where salario > 10000;

```

id_puesto	nombre	salario
1	Analog Circuit Design manager	28500.98
2	Junior Executive	10508.47
3	Director of Sales	28725.56
4	Staff Scientist	14965.31
5	Desktop Support Technician	15885.41
6	Budget/Accounting Analyst III	17131.23
7	Accounting Assistant III	29257.91
8	Programmer Analyst II	23223.95
9	Nurse Practitioner	11483.4
10	Social Worker	18008.12

```

30 • select * from articulo where precio > 1000 and iva > 100;

```

id_articulo	nombre	precio	iva	cantidad
2	Pasta - Angel Hair	4391.73	959.51	503
3	Soup Campbells - Tomato Bisque	2991.35	587.59	604
4	Wine - Valpolicella Masi	2625.2	770.1	575
5	Mousse - Banana Chocolate	3701.62	893.46	248
7	Nantucket - Kiwi Berry Cktl.	5579.47	1012.33	527
8	Wine - Fontanafredda Barolo	2684.64	327.16	682
9	Lotus Rootlets - Canned	1996.46	324.72	636
10	Wine - Vovray Sec Domaine Huet	6066.99	890.47	397
11	Cake - Pancake	5271.11	821.28	64
12	Chocolate Liqueur - Godet White	1616.78	612.63	929

```

31
32 • select * from venta where (id_articulo = 135 or id_articulo = 963) and (id_empleado = 835 or id_empleado = 369);

```

id_venta	id_articulo	id_empleado	clave	fecha
7	963	369	47335-894	2019-06-08 00:00:00
6	135	835	0049-0032	2020-02-03 15:05:27
NULL	NULL	NULL	NULL	NULL

```

44
45 • select * from puesto order by salario desc limit 5 ;

```

id_puesto	nombre	salario
494	Sales Representative	29996.58
18	Speech Pathologist	29967.17
487	Analog Circuit Design manager	29923.95
79	Junior Executive	29916.06
893	Technical Writer	29912.53
NULL	NULL	NULL

Sesión 2

Clase 6/07/2020

Reto 1

Agrupamientos y subconsultas

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

`select * from articulo where nombre like '%Pasta%';`

	id_articulo	nombre	precio	iva	cantidad
▶	2	Pasta - Angel Hair	4391.73	959.51	503
	27	Pasta - Elbows, Macaroni, Dry	3668.7	253.66	392
	70	Pasta - Shells, Medium, Dry	801.74	773.8	206
	91	Pasta - Cheese / Spinach Bauletti	5811.44	619.36	15
	134	Pasta - Orzo, Dry	6537.81	1113.88	806

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

`select * from articulo where nombre like '%Cannelloni%';`

	id_articulo	nombre	precio	iva	cantidad
▶	233	Pasta - Cannelloni, Sheets, Fresh	2316.37	605.55	307
*	NULL	NULL	NULL	NULL	NULL

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

`select * from articulo where nombre like '%-%';`

	id_articulo	nombre	precio	iva	cantidad
▶	1	Chocolate - Feathers	2738.93	12.26	144
	2	Pasta - Angel Hair	4391.73	959.51	503
	3	Soup Campbells - Tomato Bisque	2991.35	587.59	604
	4	Wine - Valpolicella Masi	2625.2	770.1	575
	5	Mousse - Banana Chocolate	3701.62	893.46	248

Reto 2

Agrupamientos y subconsultas

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

`select avg(salario) from puesto;`

	avg(salario)
▶	19595.051179999973

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

select count() from articulo where nombre like '%Pasta%';*

	count(*)
▶	17

- ¿Cuál es el salario mínimo y máximo?

select min(salario) as SalarioMaximo, max(salario) as SalarioMinimo from puesto;

	SalarioMaximo	SalarioMinimo
▶	10013.44	29996.58

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

select sum(salario) as Suma from puesto where id_puesto >= 995;

	Suma
▶	98919.69

Reto 3

Agrupamientos

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

select nombre, count() as TotalPuestos from puesto group by nombre;*

	nombre	TotalPuestos
▶	Analog Circuit Design manager	8
	Junior Executive	8
	Director of Sales	8
	Staff Scientist	9
	Desktop Support Technician	5
	Budget/Accounting Analyst III	4

- ¿Cuánto dinero se paga en total por puesto?

select nombre, sum(salario) as SalarioTotalPuesto from puesto group by nombre;

	nombre	SalarioTotalPuesto
	Staff Scientist	157528.98
	Desktop Support Technician	92315.22
	Budget/Accounting Analyst III	70107.77
	Accounting Assistant III	78947.08
▶	Programmer Analyst II	35658.78
	Nurse Practitioner	296384.04
	Social Worker	168107.81
	Teacher	175127.00000000007

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

select id_empleado, count() as TotalVentas from venta group by id_empleado;*

	id_empleado	TotalVentas
▶	2	2
	3	2
	4	1
	5	1
	6	2
	12	5

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

select id_articulo, count() as TotalVentasArticulo from venta group by id_articulo;*

	id_articulo	TotalVentasArticulo
▶	2	1
	3	1
	4	2
	8	1
	10	1
	11	1

Reto 4

Subconsultas

- ¿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);

	nombre
--	--------

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

select id_empleado, min(total_ventas), max(total_ventas) from (Select clave, id_empleado, count() total_ventas from venta group by clave, id_empleado) as sq group by id_empleado;*

	id_empleado	min(total_ventas)	max(total_ventas)
▶	569	1	1
	413	1	2
	765	1	1
	119	1	1
	--	.	.

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

select nombre, apellido_paterno, (select nombre from puesto p where id_puesto = e.id_puesto) as Puesto from empleado as e;

	nombre	apellido_paterno	Puesto
▶	Enrichetta	Bodechon	Product Engineer
	Morey	Bowskill	Budget/Accounting Analyst IV
	Jeannette	Potes	Occupational Therapist
	Cassey	Womersley	Financial Advisor
	Gnni	Risom	Physical Therapy Assistant
	Lisle	Carlsson	Marketing Assistant
	Andee	Thomson	Team Assistant

Proyecto Sesión 2

Dentro de la tabla `employees`, obten el número de empleado, apellido y nombre de todos los empleados cuyo nombre empieza con 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 'a%';

	employeeNumber	lastname	firstName
▶	1143	Bow	Anthony
	1611	Fixter	Andy
*	NULL	NULL	NULL

2. 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';

	employeeNumber	lastname	firstName
*	NULL	NULL	NULL

3. 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';

	employeeNumber	lastname	firstName
▶	1619	King	Tom
*	NULL	NULL	NULL

4. 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%';

	employeeNumber	lastname	firstName
▶	1611	Fixter	Andy
	1143	Bow	Anthony
	1002	Murphy	Diane
	1286	Tseng	Foon Yue
	1323	Vanauf	George
	1102	Bondur	Gerard
	1370	Hernandez	Gerard

5. 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%';

	productCode	productName
▶	S10_2016	1996 Moto Guzzi 1100i
	S18_3320	1917 Maxwell Touring Car
	S24_2000	1960 BSA Gold Star DBD34
	S24_2011	18th century schooner
	S24_2022	1938 Cadillac V-16 Presidential Limousine
	S24_3420	1937 Horch 930V Limousine
	S24_4620	1961 Chevrolet Impala
	S32_2206	1982 Ducati 996 R
	S32_3207	1950's Chicago Surface Lines Streetcar
	S700_2047	HMS Bounty
	NULL	NULL

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

select orderNumber, sum(priceEach) as total from orderdetails group by orderNumber;

	orderNumber	total
▶	10100	301.84
	10101	352.00
	10102	138.68
	10103	1520.37
	10104	1251.89
	10105	1479.71
	10106	1427.28

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

select count(orderNumber), year(orderDate) as año from orders group by año;

	count(orderNumber)	año
▶	111	2003
	151	2004
	64	2005

8. 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 as O where country = 'USA');

	lastName	firstName
▶	Murphy	Diane
	Patterson	Mary
	Firrelli	Jeff
	Bow	Anthony
	Jennings	Leslie
	Thompson	Leslie
	Firrelli	Julie
	Patterson	Steve
	Tseng	Foon Yue
	Vanauf	George

9. *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 = (select max(amount) from payments) ;

	customerNumber	checkNumber	amount
▶	141	JE105477	120166.58
*	NULL	NULL	NULL

10. *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 max(amount) from payments) ;

	customerNumber	checkNumber	amount ▲
	172	EH208589	33383.14
	202	IQ627690	33594.58
	242	AF40894	33818.34
	128	FH668230	33820.62
	249	IJ399820	33924.24
	398	AJ478695	33967.73
	186	KA602407	34341.08
	339	DA98827	34606.28
	121	MA302151	34638.14

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

select customerNumber, customername from customers where

customerNumber not in (select customerNumber from orders);

	customerNumber	customername
▶	125	Havel & Zbyszek Co
	168	American Souvenirs Inc
	169	Porto Imports Co.
	206	Asian Shopping Network, Co
	223	Natürlich Autos
	237	ANG Resellers
	247	Messner Shopping Network
	273	Franken Gifts, Co
	293	BG&E Collectables
	303	Schuyler Imports
	307	Der Hund Imports

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

*select max(quantityOrdered), min(quantityOrdered), avg(quantityOrdered)
from orderdetails;*

	max(quantityOrdered)	min(quantityOrdered)	avg(quantityOrdered)
▶	97	6	35.2190

- 13.** *Dentro de la tabla orders, obten el número de órdenes que hay por cada estado.*

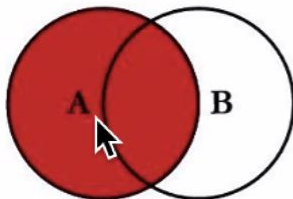
*select state, sum((select count(orderNumber) from orders as o where
o.customerNumber = c.customerNumber)) as NOrder from customers as c
group by state;*

	state	NOrder
▶	NULL	180
	CA	45
	MA	23
	NY	18
	PA	9
	Victoria	8
	CT	8
	NSW	8
	BC	4

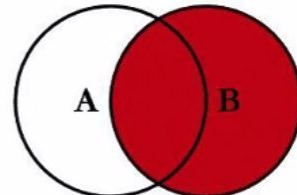
Sesión 3

Clase 8/07/2020

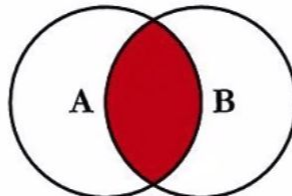
SQL JOINS



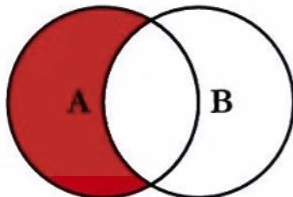
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



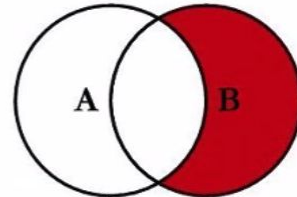
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



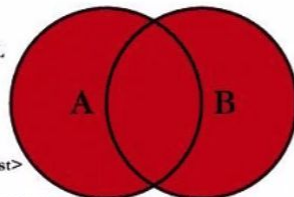
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



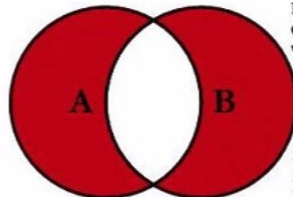
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

Reto 1

Relaciones

- ¿Cuál es el nombre de los empleados que realizaron cada venta?
select nombre from empleado e inner join venta v on e.id_empleado = v.id_empleado;

	nombre
►	Morey
	Morey
	Jeannette
	Jeannette
	Cassey
	Gnni
	Lisle
	Lisle

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

```
select distinct v.id_venta, a.nombre from articulo a inner join venta v on
a.id_articulo = v.id_articulo;
```

	id_venta	nombre
▶	919	Pasta - Angel Hair
	885	Soup Campbells - Tomato Bisque
	473	Wine - Valpolicella Masi
	504	Wine - Valpolicella Masi
	387	Wine - Fontanafredda Barolo
	629	Wine - Vovray Sec Domaine Huet
	845	Cake - Pancake
	288	Chocolate Liqueur - Godet White
	144	Appetizer - Southwestern

- ¿Cuál es el total de cada venta?

```
SELECT v.clave,sum(a.precio) as total from venta as v join articulo as a on
v.id_articulo = a.id_articulo GROUP BY v.clave;
```

	clave	total
▶	0228-3661	3714.37
	52125-277	340582.64999999999
	0049-0032	9999999
	13107-062	243071.43999999993
	47335-894	223650.32000000007
	51655-951	190821.20999999996
	52380-1865	162361.13
	69128-001	174310.76
	52343-028	150970.75

Reto 2

Definición de vistas

- Obtener el puesto de un empleado.

```
create view MMPuestoEmpleado as (select e.nombre as empleado, p.nombre
as puesto from empleado e join puesto p on e.id_puesto = p.id_puesto);
```

60 20:59:24 create view MMPuestoEmpleado as (select e.nombre as empleado, p.nombre as pue...

- Saber qué artículos ha vendido cada empleado.

```
create view MMEpleadoArticulos as (select e.nombre as empelado,
count(v.id_articulo) as TotalArticulos from empleado e join venta v on
e.id_empleado = v.id_empleado group by e.nombre);
```

71 21:07:04 create view MMEpleadoArticulos as (select e.nombre as empelado, count(v.id_artic... 0 row(s) affected

- Saber qué puesto ha tenido más ventas

```
create view MMTotalVentEmpl as (select p.nombre as Puesto,
count(v.id_venta) as TotalVentas from puesto p join empleado e on
p.id_puesto = e.id_puesto join venta v on e.id_empleado = v.id_empleado
```

group by p.nombre order by TotalVentas desc limit 1);

✓ 90 21:13:29 create view MMTotVentEmpl as (select p.nombre as Puesto, count(v.id_venta) as ... 0 row(s) affected

Proyecto Sesión 3

Sesión 4

Clase 13/07/2020

En Mongo DB se trabaja con Json y Json trabaja con documentos (con arreglos).

Base de datos = Colecciones .

Colecciones tienen registros y tipos de datos descendente -1 ascendente 1

un registro puede tener arreglo

FILTER	{ \$and: [{ year: { \$gte: 2012 } }, { year: { \$lte: 2019 } }] }
PROJECT	{title:1,year:1}
SORT	{Year:-1}
COLLATION	

filter es un where

project es en lugar de un asterisco

sort funciona como un order

collation

Reto 1

Reto 1: Colecciones, Documentos y Proyecciones

- Fecha, nombre y texto de cada comentario.

```
{ date :1, name : 1,text:1, _id:0}
```

```
name: "Andrea Le"
text: "Rem officiis eaque repellendus amet eos doloribus. Porro dolor volupta..."
date: 2012-03-26T23:20:16.000+00:00
```

```
name: "Greg Powell"
text: "Tenetur dolorum molestiae ea. Eligendi praesentium unde quod porro. Co..."
date: 1987-02-10T00:29:36.000+00:00
```

- Título, elenco y año de cada película.

```
{title:1, cast:1,year:1,_id:0}
```

```
_id: ObjectId("573a1390f29313caabcd4135")
cast: Array
  0: "Charles Kayser"
  1: "John Ott"
title: "Blacksmith Scene"
year: 1893
```

```
_id: ObjectId("573a1390f29313caabcd42e8")
cast: Array
title: "The Great Train Robbery"
year: 1903
```

- Nombre y contraseña de cada usuario.

```
{name:1,password:1,_id:0}
```

Reto 2: Filtros básicos

- ¿Qué comentarios ha hecho Greg Powell?

```
Filter {name: "Greg Powell"}
```






```
_id: ObjectId("5a9427648b0beeb69579cf")
name: "Greg Powell"
email: "greg_powell@fakegmail.com"
movie_id: ObjectId("573a1390f29313caabcd41b1")
text: "Tenetur dolorum molestiae ea. Eligendi praesentium unde quod porro. Co..."
date: 1987-02-10T00:29:36.000+00:00
```

- ¿Qué comentarios han hecho Greg Powell o Mercedes Tyler?
Filter {\$or: [{ name: "Greg Powell" } , { name : "Mercedes Tyler"}] }

```
_id: ObjectId("5a9427648b0beeb69579cf")
name: "Greg Powell"
email: "greg_powell@fakegmail.com"
movie_id: ObjectId("573a1390f29313caab41b1")
text: "Tenetur dolorum molestiae ea. Eligendi praesentium unde quod porro. Co..."
date: 1987-02-10T00:29:36.000+00:00
```

```
_id: ObjectId("5a9427648b0beeb69579e7")
name: "Mercedes Tyler"
email: "mercedes_tyler@fakegmail.com"
movie_id: ObjectId("573a1390f29313caab4323")
text: "Eius veritatis vero facilis quaerat fuga temporibus. Praesentium exped..."
date: 2002-08-18T04:56:07.000+00:00
```

- ¿Cuál es el máximo número de comentarios en una película?
sort {num_mflix_comments:-1}



VIEW




```
_id: ObjectId("573a1399f29313caabcee886")
fullplot: "Stanley Ipkiss is a bank clerk that is an incredibly nice man. Unfortu..."
> imdb: Object
year: 1994
plot: "Bank clerk Stanley Ipkiss is transformed into a manic super-hero when ..."
> genres: Array
rated: "PG-13"
metacritic: 56
title: "The Mask"
lastupdated: "2015-09-04 00:22:53.413000000"
> languages: Array
> writers: Array
type: "movie"
> tomatoes: Object
poster: "https://m.media-amazon.com/images/M/MV5BOWExYjI5MzktNTRhNi00ZWZg2LTkZm..."
num_mflix_comments: 456
released: 1994-07-29T00:00:00.000+00:00
> awards: Object
> countries: Array
> cast: Array
> directors: Array
runtime: 101
```

- ¿Cuál es título de las cinco películas más comentadas?

Project {title:1,num_mflix_comments:1,_id:0}
Sort {num_mflix_comments:-1 }

<div><div><div><div><div></div><div>VIEW</div></div><div><div></div><div>{ }</div><div></div></div></div></div></div>
<div><div>title: "The Mask"</div><div>num_mflix_comments: 456</div></div>
<div><div>title: "Dumb & Dumber"</div><div>num_mflix_comments: 450</div></div>
<div><div>title: "The Unborn"</div><div>num_mflix_comments: 447</div></div>
<div><div>title: "About a Boy"</div><div>num_mflix_comments: 441</div></div>
<div><div>title: "8 Mile"</div><div>num_mflix_comments: 441</div></div>

Proyecto Sesión 4

1. Obtén los datos de contacto de cada compañía.

Project : {name:1,email_address:1, phone_number:1, description:1 ,_id:0}

```
{
  name: 1,
  email_address: 1,
  phone_number: 1,
  description: 1,
  _id: 0
}
```

```
}
{
  name: "Wetpaint"
  email_address: "info@wetpaint.com"
  phone_number: "206.859.6300"
  description: "Technology Platform Company"
}
{
  name: "Facebook"
  email_address: ""
  phone_number: ""
  description: "Social network"
}
{
  name: "Omnidrive"
  email_address: "info@omnidrive.com"
  phone_number: "660-675-5052"
  description: null
}
}
```

- Obtén la fuente de cada tweet.

Project: {text:1,source:1,created_at:1,_id:0}

```
{
  text: 1,
  source: 1,
  created_at: 1,
  _id: 0
}
```

```
{
  text: "eu preciso de terminar de fazer a minha tabela, está muito foda **"
```

```
created_at: "Thu Sep 02 18:11:23 +0000 2010"
```

```
source: "web"
```

```

  text: "I can't wait for #BoardwalkEmpire"
```

```
created_at: "Thu Sep 02 18:11:24 +0000 2010"
```

```
source: "<a href='\"http://www.tweetdeck.com\"' rel='\"nofollow\"'>TweetDeck</a>"
```

```

  text: "Oky nenek nya RT @wikigehol: Oky jd anak na yyyy RT @okyoktaaaaa: Papa..."
```

```
created_at: "Thu Sep 02 18:11:24 +0000 2010"
```

```
source: "<a href='\"http://blackberry.com/twitter\"' rel='\"nofollow\"'>Twitter for Bla..."
```

- Obtén el nombre de todas las compañías fundadas en octubre. Y
- Obtén el nombre de todas las compañías fundadas en 2008.

Filter : {\$and: [{ founded_year: 2008},{ founded_month: 10}]}

project: {name:1,founded_month:1,founded_year:1,_id:0}

```
{
  filter: {
    $and: [
      {
        founded_year: 2008
```

```

    },
    {
      founded_month: 10
    }
  ]
},
project: {
  name: 1,
  founded_month: 1,
  founded_year: 1,
  _id: 0
}
}

```

<pre> name: "tunesBag" founded_year: 2008 founded_month: 10 </pre>
<pre> name: "Muecs" founded_year: 2008 founded_month: 10 </pre>
<pre> name: "Rush Hour" founded_year: 2008 founded_month: 10 </pre>
<pre> name: "OUTSHOUTS" founded_year: 2008 founded_month: 10 </pre>

5. Obtén todos los post del autor machine.

```

Filter: {author:"machine"}
Project {_id:0}{
  filter: {
    author: 'machine'
  },
  project: {
    _id: 0
  }
}

```

}

```

body: "Amendment I
      <p>Congress shall make no law respecting an establishment ..."
permalink: "aRjNnLZkJKtyspAIoRGe"
author: "machine"
title: "Bill of Rights"
tags: Array
comments: Array
date: 2012-11-20T05:05:15.231+00:00

```

```

body: "We the People of the United States, in Order to form a more p
permalink: "jNsgObovWyKEoXNydTis"
author: "machine"
title: "US Constitution"
tags: Array
comments: Array
date: 2012-11-20T05:05:15.232+00:00

```

6. Obtén todos los tweets provenientes de la web.

Filter {source: "web"}

Project {text:1,source:1,_id:0}

```

{
  filter: {
    source: 'web'
  },
  project: {
    text: 1,
    source: 1,
    _id: 0
  }
}

```

```

text: "eu preciso de terminar de fazer a minha tabela, está muito f
source: "web"

```

```

text: "First week of school is over :P"
source: "web"

```

```

text: "fair today!!!! then jersey shore!!!=D"
source: "web"

```

```

text: "@teetolegit 1mfao!! No BS! hahaha"
source: "web"

```

7. Obtén todas las compañías fundadas en octubre del 2008.

```
Filter : {$and: [{ founded_year: 2008},{ founded_month: 10}]}{
project: {name:1,founded_month:1,founded_year:1,_id:0}
{
filter: {
$and: [
{
founded_year: 2008
},
{
founded_month: 10
}
]
},
project: {
name: 1,
founded_month: 1,
founded_year: 1,
_id: 0
}
}
```

<pre>name: "tunes8ag" founded_year: 2008 founded_month: 10</pre>
<pre>name: "Muecs" founded_year: 2008 founded_month: 10</pre>
<pre>name: "Rush Hour" founded_year: 2008 founded_month: 10</pre>
<pre>name: "OUTSHOUTS" founded_year: 2008 founded_month: 10</pre>

8. Obtén todas las compañías con más de 50 empleados.

```
Filter {number_of_employees: {$gt: 50}}
Project {name: 1,number_of_employees: 1,_id: 0} {
filter: {
number_of_employees: {
$gt: 50
```

```

    }
  },
  project: {
    name: 1,
    number_of_employees: 1,
    _id: 0
  }
}

```

```

name: "Facebook"
number_of_employees: 5299

```

```

name: "Twitter"
number_of_employees: 1300

```

```

name: "Cisco"
number_of_employees: 63000

```

```

name: "Yahoo!"
number_of_employees: 13600

```

```

name: "Powerset"
number_of_employees: 60

```

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

```
Filter {$and: [{comments:{$gte:10}}, {comments:{$lte:30}}]}
```

```
Project {description:1,comments:1,_id:0}
```

```

{
  filter: {
    $and: [
      {
        comments: {
          $gte: 10
        }
      },
      {
        comments: {
          $lte: 30
        }
      }
    ]
  },
  project: {

```



```

description: 1,
comments: 1,
_id: 0
}
}

```

```

comments: 15
description: "Treehouses bring us closer to nature, and appeal to the kid in all of ..."

```

```

comments: 14
description: "This image of the open star cluster NGC 7380, also known as the Wizard..."

```

```

comments: 12
description: "A new infrared image from NASA's Wide-field Infrared Survey Explorer, ..."

```

```

comments: 22
description: "Heder stars as a man chasing big dreams – even as he loses his job and..."

```

```

comments: 23
description: "A whole range of activities that people tend to think will make them h..."

```

10. 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,_id:0}
Sort: {number_of_employees:1}
{
filter: {
$and: [
{
number_of_employees: {
$ne: null
}
},
{
number_of_employees: {
$ne: 0
}
}
]
},

```

```

project: {
  name: 1,
  number_of_employees: 1, _id: 0
},
sort: {
  number_of_employees: 1
}
}

```

<pre> name: "FeVote" number_of_employees: 1 </pre>
<pre> name: "OurStage" number_of_employees: 1 </pre>
<pre> name: "Entrecard" number_of_employees: 1 </pre>
<pre> name: "Localeze" number_of_employees: 1 </pre>

11. Obtén la empresa con el mayor número de empleados.

```

filter {number_of_employees:{ $ne: null}}
Project {name:1,number_of_employees:1,_id:0}
{number_of_employees:-1} Limit 1
{
  filter: {
    number_of_employees: {
      $ne: null
    }
  },
  project: {
    name: 1,
    number_of_employees: 1,
    _id: 0
  },
  sort: {
    number_of_employees: -1
  },
  limit: 1
}

```

```
name: "IBM"
number_of_employees: 388000
```

12. Obtén la historia más comentada.

Project {title:1,comments:1,_id:0}

Sort {comments:-1} Limit 1

```
{
  project: {
    title: 1,
    comments: 1,
    _id: 0
  },
  sort: {
    comments: -1
  },
  limit: 1
}
```

```
title: "Republican Brown wins Massachusetts Senate seat!"
comments: 1864
```

13. Obtén la historia menos comentada.

Project {title:1,comments:1,_id:0}

Sort {comments:1} Limit 1

```
{
  project: {
    title: 1,
    comments: 1,
    _id: 0
  },
  sort: {
    comments: 1
  },
  limit: 1
}
```

```
title: "UA Tech Park chosen for $32 million 'Solar Zone' project"
comments: 0
```

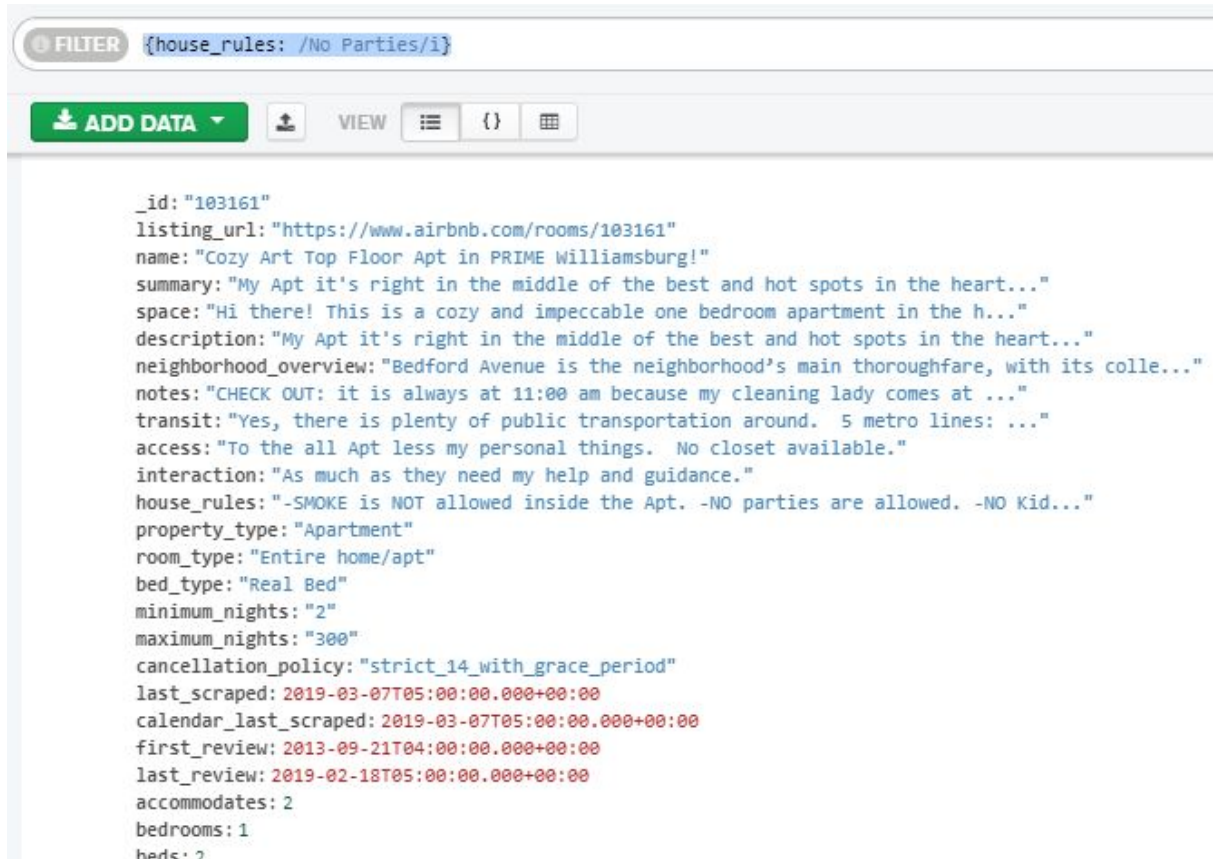
Sesión 5

Clase 15/07/2020

Reto 1: Expresiones regulares

- Propiedades que no permitan fiestas.

{house_rules: /No Parties/i}



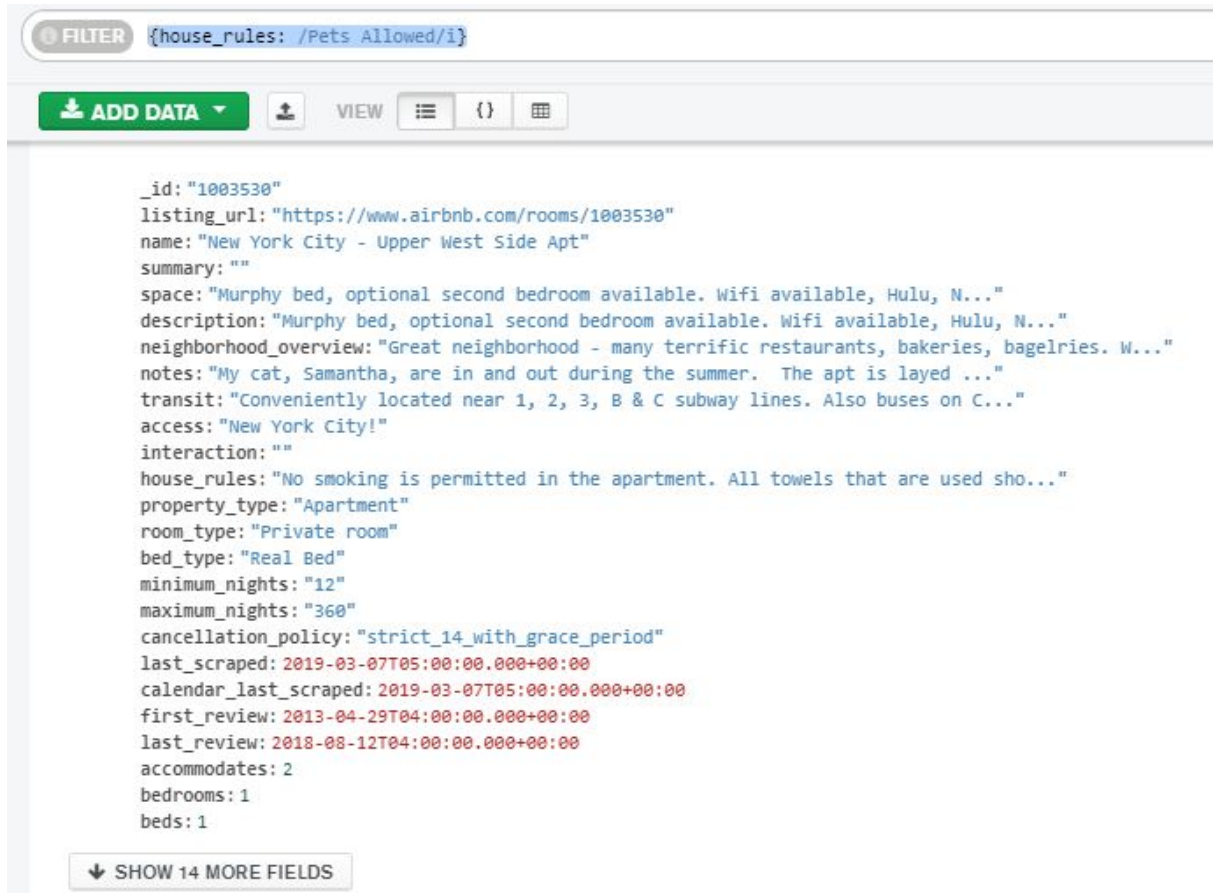
```

_id: "103161"
listing_url: "https://www.airbnb.com/rooms/103161"
name: "Cozy Art Top Floor Apt in PRIME Williamsburg!"
summary: "My Apt it's right in the middle of the best and hot spots in the heart..."
space: "Hi there! This is a cozy and impeccable one bedroom apartment in the h..."
description: "My Apt it's right in the middle of the best and hot spots in the heart..."
neighborhood_overview: "Bedford Avenue is the neighborhood's main thoroughfare, with its colle..."
notes: "CHECK OUT: it is always at 11:00 am because my cleaning lady comes at ..."
transit: "Yes, there is plenty of public transportation around. 5 metro lines: ..."
access: "To the all Apt less my personal things. No closet available."
interaction: "As much as they need my help and guidance."
house_rules: "-SMOKE is NOT allowed inside the Apt. -NO parties are allowed. -NO Kid..."
property_type: "Apartment"
room_type: "Entire home/apt"
bed_type: "Real Bed"
minimum_nights: "2"
maximum_nights: "300"
cancellation_policy: "strict_14_with_grace_period"
last_scraped: 2019-03-07T05:00:00.000+00:00
calendar_last_scraped: 2019-03-07T05:00:00.000+00:00
first_review: 2013-09-21T04:00:00.000+00:00
last_review: 2019-02-18T05:00:00.000+00:00
accommodates: 2
bedrooms: 1
beds: 2

```

- Propiedades que admitan mascotas.

{house_rules: /Pets Allowed/i}



FILTER {house_rules: /Pets Allowed/i}

ADD DATA VIEW

```

_id: "1003530"
listing_url: "https://www.airbnb.com/rooms/1003530"
name: "New York City - Upper West Side Apt"
summary: ""
space: "Murphy bed, optional second bedroom available. Wifi available, Hulu, N..."
description: "Murphy bed, optional second bedroom available. Wifi available, Hulu, N..."
neighborhood_overview: "Great neighborhood - many terrific restaurants, bakeries, bagelries. W..."
notes: "My cat, Samantha, are in and out during the summer. The apt is layed ..."
transit: "Conveniently located near 1, 2, 3, B & C subway lines. Also buses on C..."
access: "New York City!"
interaction: ""
house_rules: "No smoking is permitted in the apartment. All towels that are used sho..."
property_type: "Apartment"
room_type: "Private room"
bed_type: "Real Bed"
minimum_nights: "12"
maximum_nights: "360"
cancellation_policy: "strict_14_with_grace_period"
last_scraped: 2019-03-07T05:00:00.000+00:00
calendar_last_scraped: 2019-03-07T05:00:00.000+00:00
first_review: 2013-04-29T04:00:00.000+00:00
last_review: 2018-08-12T04:00:00.000+00:00
accommodates: 2
bedrooms: 1
beds: 1

```

↓ SHOW 14 MORE FIELDS

- Propiedades que no permitan fumadores.

```
{house_rules: /No Smoking/i}
```

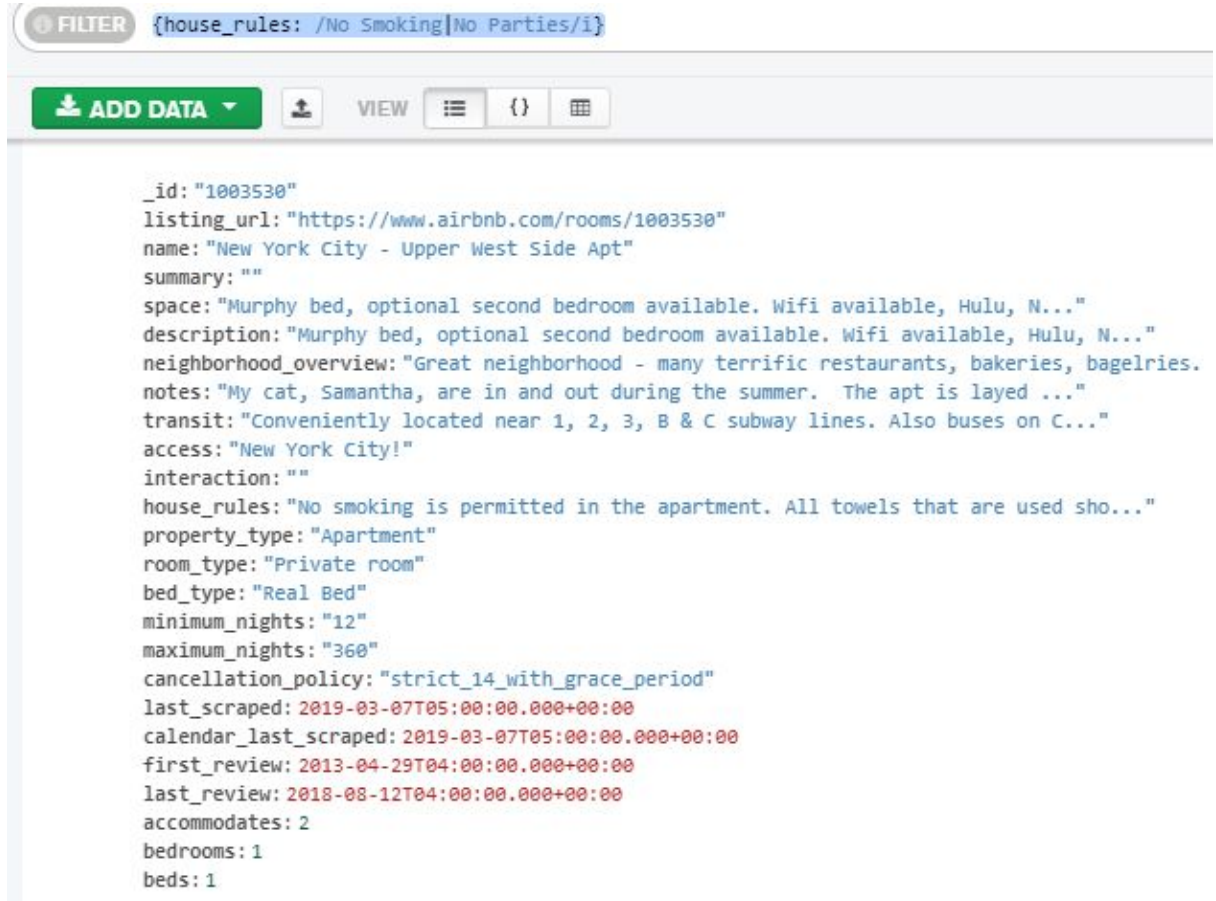
The screenshot shows a web application interface with a search filter bar at the top containing the text `{house_rules: /No Smoking/i}`. Below the filter bar is a toolbar with buttons for 'ADD DATA', 'VIEW', and icons for list, JSON, and table views. The main content area displays a JSON object representing an Airbnb listing. The JSON object includes fields such as `_id`, `listing_url`, `name`, `summary`, `space`, `description`, `neighborhood_overview`, `notes`, `transit`, `access`, `interaction`, `house_rules`, `property_type`, `room_type`, `bed_type`, `minimum_nights`, `maximum_nights`, `cancellation_policy`, `last_scraped`, `calendar_last_scraped`, `first_review`, `last_review`, `accommodates`, `bedrooms`, and `beds`. The `house_rules` field contains the string "No smoking is permitted in the apartment. All towels that are used sho...". At the bottom of the JSON object, there is a button labeled 'SHOW 14 MORE FIELDS'.

```
{
  "_id": "1003530",
  "listing_url": "https://www.airbnb.com/rooms/1003530",
  "name": "New York City - Upper West Side Apt",
  "summary": "",
  "space": "Murphy bed, optional second bedroom available. Wifi available, Hulu, N...",
  "description": "Murphy bed, optional second bedroom available. Wifi available, Hulu, N...",
  "neighborhood_overview": "Great neighborhood - many terrific restaurants, bakeries, bagelries. W...",
  "notes": "My cat, Samantha, are in and out during the summer. The apt is layed ...",
  "transit": "Conveniently located near 1, 2, 3, B & C subway lines. Also buses on C...",
  "access": "New York City!",
  "interaction": "",
  "house_rules": "No smoking is permitted in the apartment. All towels that are used sho...",
  "property_type": "Apartment",
  "room_type": "Private room",
  "bed_type": "Real Bed",
  "minimum_nights": "12",
  "maximum_nights": "360",
  "cancellation_policy": "strict_14_with_grace_period",
  "last_scraped": "2019-03-07T05:00:00.000+00:00",
  "calendar_last_scraped": "2019-03-07T05:00:00.000+00:00",
  "first_review": "2013-04-29T04:00:00.000+00:00",
  "last_review": "2018-08-12T04:00:00.000+00:00",
  "accommodates": 2,
  "bedrooms": 1,
  "beds": 1
}
```

SHOW 14 MORE FIELDS

- Propiedades que no permitan fiestas ni fumadores.

```
{house_rules: /No Smoking|No Parties/i}
```



Reto 2: Introducción a las agregaciones

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.

```

{
  filter: {
    number_of_reviews: {
      $gte: 50
    },
    'review_scores.review_scores_rating': {
      $gte: 80
    },
    amenities: {
      $in: [
        RegExp('Ethernet')
      ]
    }
  }
}

```

```

},
'address.country_code': 'BR'
}
}

```

```

>
_id: "1063491"
listing_url: "https://www.airbnb.com/rooms/1063491"
name: "Charming Apartment,perfect Location"
summary: "Charmoso quarto e sala, privativo, finamente decorado, com uma delicio..."
space: "It is a bedroom and living room, charming, private, finely decorated, ..."
description: "It is a bedroom and living room, charming, private, finely decorated, ..."
neighborhood_overview: "There are a lot of bars, restaurants, supermarkets, bakery, banks... I..."
notes: "1) Enjoy the apartment 2) Enjoy Rio 3) Come back soon"
transit: "Buses, taxis, metro and bicicle, that you can rent near to the buildin..."
access: "The full apartment"
interaction: "I like to do some programs with my guests. It depends on my time and i..."
house_rules: "1.0 apartamento situa-se em edifício residencial. Não é permitido faze..."
property_type: "Apartment"
room_type: "Entire home/apt"
bed_type: "Real Bed"
minimum_nights: "2"
maximum_nights: "89"
cancellation_policy: "super_strict_30"
last_scraped: 2019-02-11T05:00:00.000+00:00
calendar_last_scraped: 2019-02-11T05:00:00.000+00:00
first_review: 2014-01-14T05:00:00.000+00:00
last_review: 2019-01-02T05:00:00.000+00:00
accommodates: 4
bedrooms: 1
beds: 2

```

↓ SHOW 14 MORE FIELDS

Reto 3: Introducción a las agregaciones

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).

```

[
  {
    $match: {
      amenities: {
        $in: ["Wifi", "Ethernet"]
      }
    }
  }, {
    $group: {
      _id: null,
      total: {

```

```

    $sum: 1
  }
}

```

The screenshot shows the MongoDB Compass interface with two stages of an aggregation pipeline:

- \$match stage:** The input document is:


```

1 {
2   amenities: { $in: ["Wifi", "Ethernet"] }
3 }
4

```

 The output shows a sample of 20 documents, including details about an apartment like "space", "description", "neighborhood_overview", "notes", and "transit".
- \$group stage:** The input document is:


```

1 /**
2  * _id: The id of the group.
3  * fieldN: The first field name.
4  */
5 {
6   _id: null,
7   total: {
8     $sum: 1
9   }
10 }

```

 The output shows a sample of 1 document:


```

_id: null
total: 5303

```

Proyecto Sesión 5

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.

```

[{$match: {
  "address.country": "Brazil"
}}, {$match: {
  amenities: {$in: [/Ethernet/]}
}}, {$match: {
  "review_scores.review_scores_rating": {$gte: 80}
}}, {$match: {
  number_of_reviews: {$gte: 50}
}}]

```

Stage 1: Basic Query

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   amenities: {$in: [/Ethernet/]}
6 }
```

Output after \$match stage (Sample of 20 documents)

```
last_scraped: 2019-02-11T05:00:00.000+00:00
calendar_last_scraped: 2019-02-11T05:00:00.000+00:00
first_review: 2016-02-10T05:00:00.000+00:00
last_review: 2018-10-15T04:00:00.000+00:00
accommodates: 6
bedrooms: 2
beds: 5
```

Stage 2: Filter by Review Rating

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   "review_scores.review_scores_rating": {$gte: 80}
6 }
```

Output after \$match stage (Sample of 20 documents)

```
last_scraped: 2019-02-11T05:00:00.000+00:00
calendar_last_scraped: 2019-02-11T05:00:00.000+00:00
first_review: 2016-02-10T05:00:00.000+00:00
last_review: 2018-10-15T04:00:00.000+00:00
accommodates: 6
bedrooms: 2
beds: 5
```

Stage 3: Filter by Number of Reviews

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   number_of_reviews: {$gte: 50}
6 }
```

Output after \$match stage (Sample of 6 documents)

```
calendar_last_scraped: 2019-02-11T05:00:00.000+00:00
first_review: 2014-01-14T05:00:00.000+00:00
last_review: 2019-01-02T05:00:00.000+00:00
accommodates: 4
bedrooms: 1
beds: 2
number_of_reviews: 110
bathrooms: 1.0
amenities: Array
price: 351.00
```

Sesión 6

Clase 20/07/2020

Reto 1: Agrupamientos

```
[{$match: {
  property_type: "House",
  bedrooms: {$gte: 1}
}}, {$addFields: {
  costo_recamara: {$divide: ["$price", "$bedrooms"]}
}}, {$group: {
  _id: "$address.country",
  recamaras: {
    $sum: 1
```

```

},
total: {
  $sum: "$costo_recamara"
}
}}, {$addFields: {
  pais: "$_id",
  costo_promedio: {
    $divide: ["$total", "$recamaras"]
  }
}}, {$project: {
  _id:0,
  pais:1,
  Total:{$round:["$costo_promedio",2]}
}}

```

The screenshot shows a MongoDB Atlas pipeline editor with two stages:

- \$addFields stage:** The input JSON is:


```

1 {
2   pais: "$_id",
3   costo_promedio: {
4     $divide: ["$total", "$recamaras"]
5   }
6 }

```

 The output sample shows:


```

_id: "Turkey"
recamaras: 36
total: 7555.50
pais: "Turkey"
costo_promedio: 209.875

```
- \$project stage:** The input JSON is:


```

1 /**
2  * specifications: The fields to
3  * include or exclude.
4  */
5 {
6   _id:0,
7   pais:1,
8   Total:{$round:["$costo_promedio",2]}
9 }
10

```

 The output sample shows:


```

pais: "Hong Kong"
Total: 514.60

```

Reto 2: Asociación de colecciones

```

[{
  $lookup: {
    from: 'users',
    localField: 'email',

```

```

        foreignField: 'email',
        as: 'user'
    }
}, {
    $addFields: {
        usr_obj: {
            $arrayElemAt: [
                '$user',
                0
            ]
        }
    }
}, {
    $match: {
        usr_obj: {
            $exists: true
        }
    }
}, {
    $addFields: {
        password: '$usr_obj.password'
    }
}, {
    $project: {
        password: 1,
        email: 1,
        name: 1,
        _id: 0
    }
}]

```


☰

▼

\$match

☑

🗑️

+

```
1 {
2   usr_obj: {
3     $exists: true
4   }
5 }
```

Output after [\\$match](#) stage ⓘ (Sample of 20 documents)

```
_id: ObjectId("5a9427648b0beebe69579cc")
name: "Andrea Le"
email: "andrea_le@fakegmail.com"
movie_id: ObjectId("573a1390f29313caabcd418c")
text: "Rem officiis eaque repellendus amet eos
      doloribus. Porro dolor volupta..."
date: 2012-03-26T23:20:16.000+00:00
user: Array
usr_obj: Object
```

☰

▼

\$addFields

☑

🗑️

+

```
1 {
2   password: '$usr_obj.password'
3 }
```

Output after [\\$addFields](#) stage ⓘ (Sample of 20 documents)

```
_id: ObjectId("5a9427648b0beebe69579cc")
name: "Andrea Le"
email: "andrea_le@fakegmail.com"
movie_id: ObjectId("573a1390f29313caabcd418c")
text: "Rem officiis eaque repellendus amet eos
      doloribus. Porro dolor volupta..."
date: 2012-03-26T23:20:16.000+00:00
user: Array
usr_obj: Object
password: "2b$12$JS87HwL2y0P1E6KycbKOKx22.wsKdLts0F"
```

☰

▼

\$project

☑

🗑️

+

```
1 {
2   password: 1,
3   email: 1,
4   name: 1,
5   _id: 0
6 }
```

Output after [\\$project](#) stage ⓘ (Sample of 20 documents)

```
name: "Andrea Le"
email: "andrea_le@fakegmail.com"
password: "2b$12$JS87HwL2y0P1E6KycbKOKx22.wsKdLts0F"
```

vista

se crea en el boton de save en la flechita -> create view
la vista se agrega en la coleccion del lado derecho

sample_mflix.mmpss (view on: sample_mflix.comments) MODIFY

Documents Aggregations Schema Explain Plan Indexes

FILTER

VIEW ⌵ { } ⌲

```

name: "Meera Reed"
email: "ellie_kendrick@gameofthron.es"
password: "$2b$12$SdyXPBMdGScx6DePpPawFeOqcpwjdHAuTXaPl0mkvWeLzZk6EWti"

name: "Emily Ellis"
email: "emily_ellis@fakegmail.com"
password: "$2b$12$UuCb5RqPEgheoLlwOF/Jb.x9gpFvMD30oUwpSRKljwo8pBUmWT6eG"

name: "Petyr Baelish"
email: "aidan_gillen@gameofthron.es"
password: "$2b$12$qM.YvmiekyYY7p7phpK30icbRCDkN7ESwYAnG/o9YnfHC0Mhkmbi"

```

Reto 3: Generación de vistas

sample_airbnb.MMS_airbnb (view on: sample_airbnb.listingsAndReviews) MODIFY SOURCE

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER

VIEW ⌵ { } ⌲

```

pais: "United States"
Total: 113.73

pais: "Australia"
Total: 107.78

pais: "China"
Total: 393.00

pais: "Brazil"
Total: 348.66

```

Proyecto Sesión 6

Continuaremos trabajando con la base de datos de películas y sus comentarios.

El proyecto consiste en obtener, por país, el número de películas que hay de cada género. Un ejemplo de salida en formato de tabla sería:

```
[{
  $unwind: {
    path: '$countries'
  }
}, {
  $unwind: {
    path: '$genres'
  }
}, {
  $group: {
    _id: {
      pais: '$countries',
      genero: '$genres'
    },
    titulos: {
      $sum: 1
    }
  }
}, {
  $project: {
    pais: '$_id.pais',
    genero: '$_id.genero',
    titulos: 1,
    _id: 0
  }
}]
```

sample_mflix.MMSProy6 (view on: sample_mflix.movies) [MODIFY SOURCE](#)

Documents Aggregations Schema Explain Plan Indexes

[FILTER](#)

[VIEW](#) [LIST](#) [JSON](#) [TABLE](#)

MMSProy6

	titulos Int32	pais String	genero String
1	23	"West Germany"	"Thriller"
2	2	"Panama"	"Adventure"
3	15	"Luxembourg"	"Romance"
4	8	"Greece"	"Thriller"
5	1	"Slovenia"	"Music"

Sesión 7


Instalación MySQLServer 8.0.21

<https://dev.mysql.com/downloads/installer/> y seguir los pasos

insertar datos user

Table Data Import

Configure Import Settings

Detected file format: csv 

Encoding: utf-8

Columns:

<input checked="" type="checkbox"/> Source Column	Field Type
<input checked="" type="checkbox"/> UserID	int
<input checked="" type="checkbox"/> Gender	text
<input checked="" type="checkbox"/> Age	int
<input checked="" type="checkbox"/> Occupation	int
<input checked="" type="checkbox"/> Zip-code	text

UserID	Gender	Age	Occupation	Zip-code
1	F	1	10	48067
2	M	56	16	70072
3	M	25	15	55117
4	M	45	7	02460
5	M	25	20	55455

< Back Next > Cancel

ratings.csv movies.csv*

```

1 UserID,MovieID,Rating,Timestamp
2 1,1193,5,978300760
3 1,661,3,978302109
4 1,914,3,978301968
5 1,3408,4,978300275
6 1,2355,5,978824291
7 1,1197,3,978302268
8 1,1287,5,978302039
9 1,2804,5,978300719
10 1,594,4,978302268
11 1,919,4,978301368
12 1,595,5,978824268
13 1,938,4,978301752
14 1,2398,4,978302281
15 1,2918,4,978302124
16 1,1035,5,978301753
17 1,2791,4,978302188
18 1,2687,3,978824268
19 1,2018,4,978301777
20 1,3105,5,978301713
21 1,2797,4,978302039
22 1,2321,3,978302205
23 1,720,3,978300760
24 1,1270,5,978300055
25 1,527,5,978824195
26 1,2340,3,978300103
27 1,48,5,978824351
28 1,1097,4,978301953
29 1,1721,4,978300055
30 1,1545,4,978824139
31 1,745,3,978824268
32 1,2294,4,978824291
33 1,3186,4,978300019
34 1,1566,4,978824330
35 1,588,4,978824268
36 1,1907,4,978824330
37 1,783,4,978824291
38 1,1836,5,978300172
39 1,1022,5,978300055
40 1,2762,4,978302091
41 1,150,5,978301777
42 1,1,5,978824268
43 1,1961,5,978301590
44 1,1062,4,978301753

```

Table Data Import

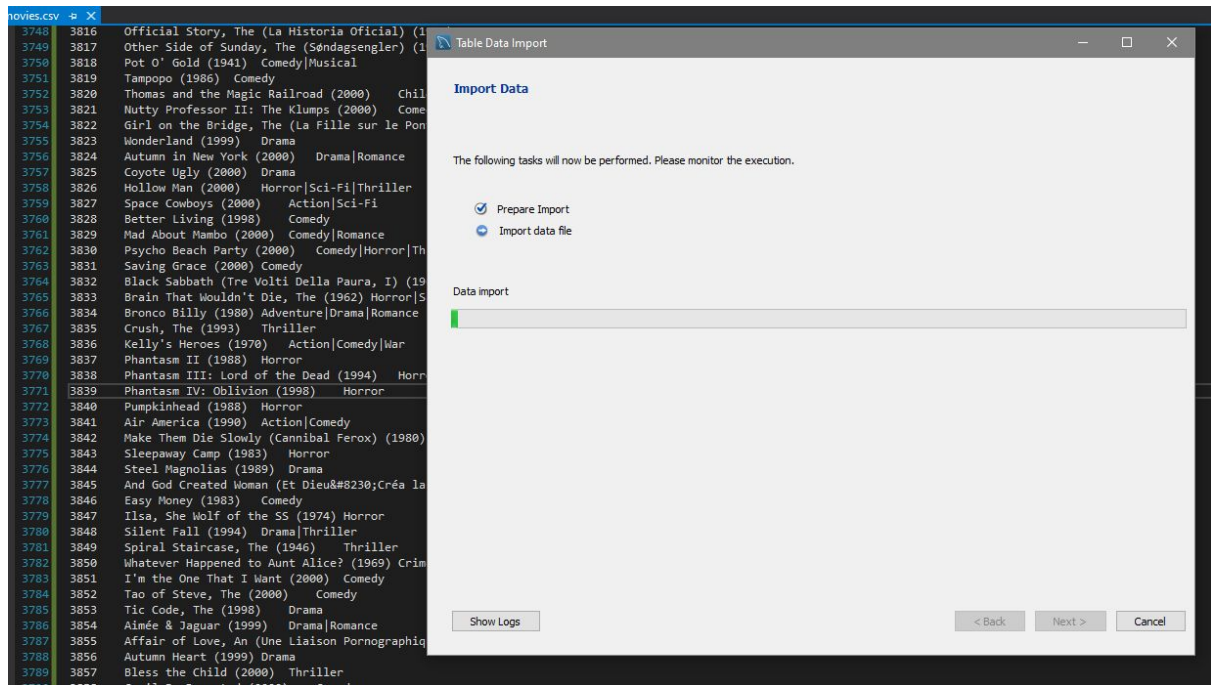
Import Data

The following tasks will now be performed. Please monitor the execution.

- ☒ Prepare Import
- ☒ Import data file

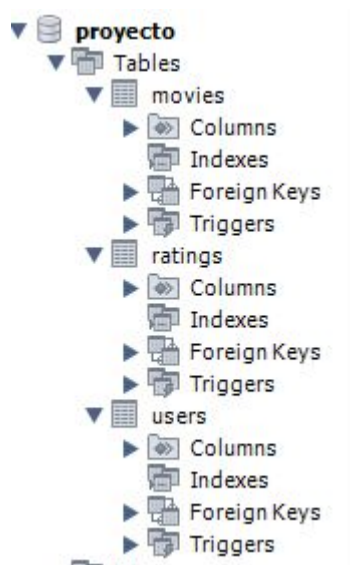
Data import

Show Logs < Back Next > Cancel

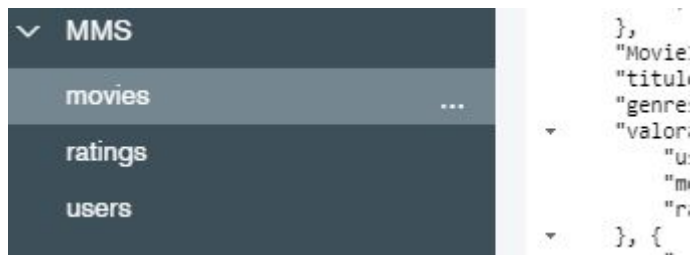


Proyecto 7

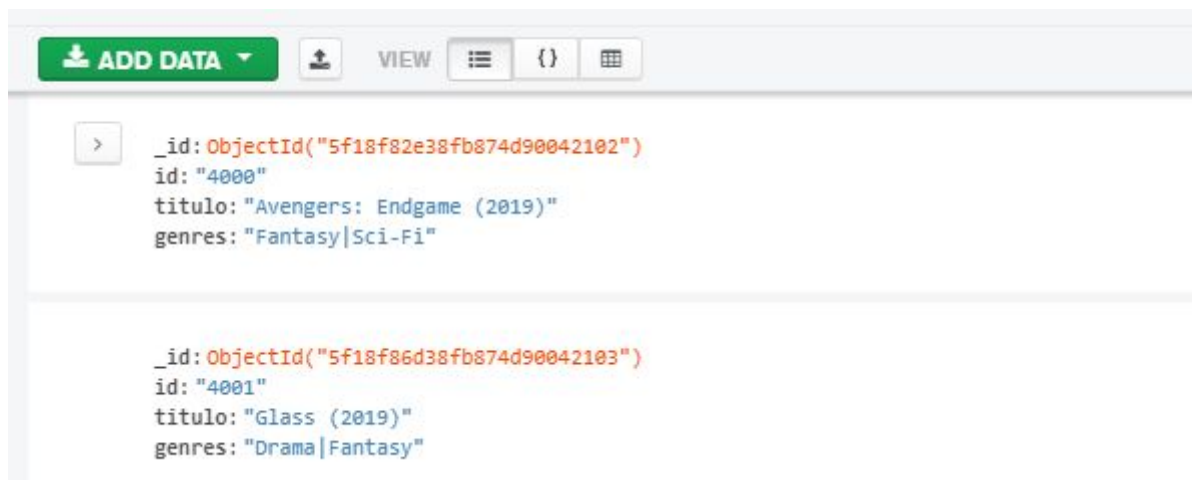
Datos en Workbench



Datos en MongoDB



insert datos



actualización información

FILTER {MovieID: "4001"} **OPTIONS** **FIND** **RESET** ...

PROJECT

SORT {MovieID :1} **MAXTIME** 5000

COLLATION **SKIP** 0 **LIMIT** 0

ADD DATA **VIEW** **{}** **REFRESH**

Displaying documents 1 - 1 of 1

```
{
  "_id": {
    "$oid": "5f18f86d38fb874d90042103"
  },
  "MovieID": "4001",
  "titulo": "Glass (2019)",
  "genres": "Drama|Fantasy",
  "valoraciones": [
    {
      "userid": "1563",
      "movieid": "4001",
      "rating": "4"
    },
    {
      "userid": "434",
      "movieid": "4001",
      "rating": "5"
    }
  ]
}
```


SESION 8

Reto 1

The screenshot shows the MongoDB Compass interface. At the top, it indicates '25600 Documents in the Collection'. The main area is divided into two panels. The left panel, titled 'Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)', contains a query editor with the following code:

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   $and:
6   [
7     {Latitude: {$gte: 19.36}},
8     {Latitude: {$lte: 19.37}},
9     {Longitude: {$gte: -99.14}},
10    {Longitude: {$lte: -98.5}}
11  ]
12 }
```

The right panel, titled 'Preview of Documents in the Collection', shows a sample document:

```
{
  "Street Address": "30101 STREET, UNIT",
  "City": "Abu Dhabi",
  "State/Province": "AZ",
  "Country": "AE",
  "Postcode": "",
  "Phone Number": "",
  "Timezone": "GMT+04:00 Asia/Dubai",
  "Longitude": 54.38,
  "Latitude": 24.48
}
```

Below the query editor, there is a section for the '\$match' stage, which is currently selected and toggled on. The output after the '\$match' stage is shown as a sample of 1 document:

```
{
  "_id": ObjectId("5f1f71248b195a3ccc206659"),
  "Brand": "Starbucks",
  "Store Number": "17835-185524",
  "Store Name": "Portal Churbusco",
  "Ownership Type": "Licensed",
  "Street Address": "Av. Rio Churubusco No.583 Local B10, C",
  "City": "Mexico City",
  "State/Province": "DFE"
}
```

Reto 2

- ¿Cuál fue el país con mayor número de muertes?
select Country, sum(Deaths) total from h1n1 where Country <> 'Grand Total'
group by Country order by total asc limit 1;
- ¿Cuál fue el país con menor número de muertes?
select Country, sum(Deaths) total from h1n1 where Country <> 'Grand Total'
group by Country order by total desc limit 1;
- ¿Cuál fue el país con el mayor número de casos?
select Country, sum(Cases) total from h1n1 where Country <> 'Grand Total'
group by Country order by total asc limit 1;
- ¿Cuál fue el país con el menor número de casos?
select Country, sum(Cases) total from h1n1 where Country <> 'Grand Total'
group by Country order by total desc limit 1;
- ¿Cuál fue el número de muertes promedio?
select Country, sum(Deaths)/count(Country) Muertespromedio from h1n1
where Country <> 'Grand Total';

- ¿Cuál fue el número de casos promedio?
select Country, sum(Cases)/count(Country) CasosPromedio from h1n1 where Country <> 'Grand Total';
- Top 5 de países con más muertes
select Country, sum(Deaths) total from h1n1 where Country <> 'Grand Total'
group by Country order by total asc limit 5;
- Top 5 de países con menos muertes
select Country, sum(Deaths) total from h1n1 where Country <> 'Grand Total'
group by Country order by total desc limit 5;

Reto 3

- ¿Cuál es país con mayor número de casos?

```

[[
  $project: {
    _id: 0,
    Region: 1,
    Confirmed: 1
  }
], {
  $group: {
    _id: '$Region',
    Region: {
      $sum: '$Confirmed'
    }
  }
}, {
  $sort: {
    Region: -1
  }
}, {
  $limit: 1
}]

```

MMS.Mayor casos (view on: MMS.COVID)



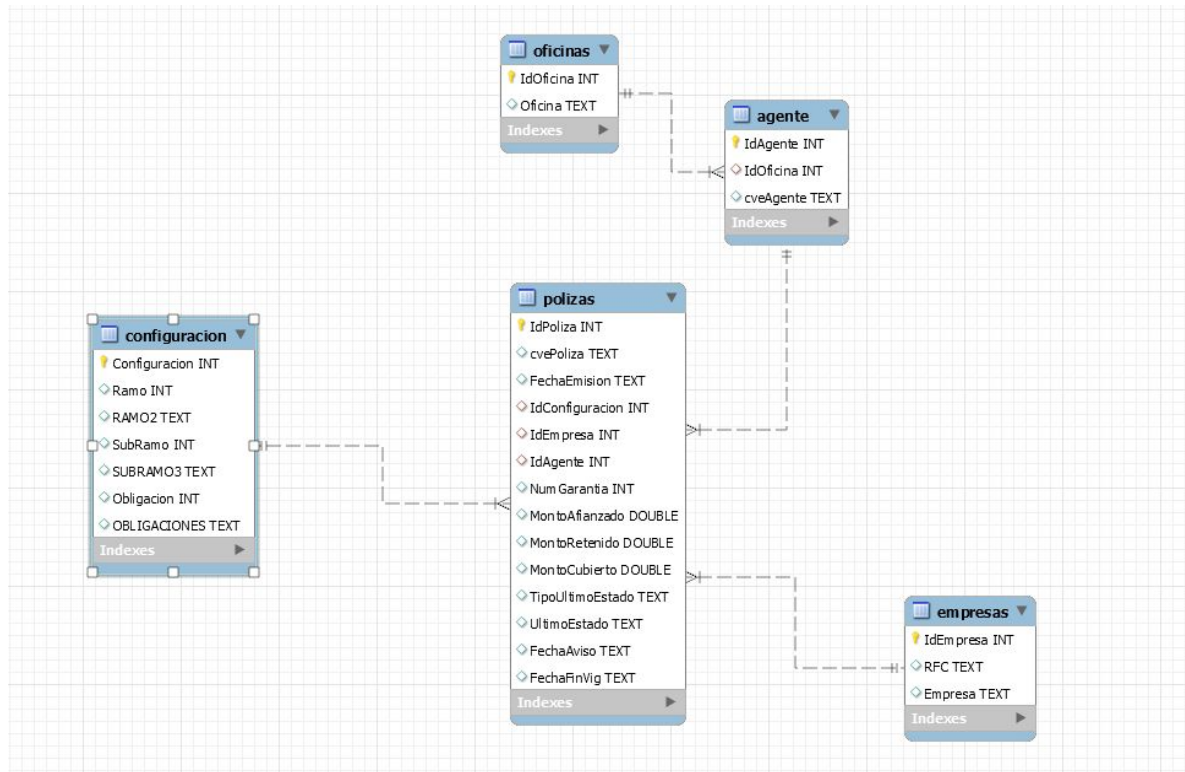
- ¿Cuál es el país con mayor número de muertes?


```
{
  $project: {
    Region: 1,
    Deaths: 1
  }
}, {
  $group: {
    _id: '$Region',
    Region: {
      $sum: '$Deaths'
    }
  }
}, {
  $sort: {
    Region: -1
  }
}, {
  $limit: 1
}]
```
- Usando las coordenadas, encuentra el epicentro del virus.
- Usando el epicentro, encuentra las 5 regiones más cercanas a dicho epicentro.

PROYECTO FINAL

Garantias

Diagrama MySQL



Preguntas

Sesión 1 - MySQL

Sesión 2 - MySQL

Sesión 3 - MySQL

Sesión 4 - MongoDB

Sesión 5 - MongoDB

Sesión 6 - MongoDB