

# SESIÓN 01

## RETO 1

**01. Usando la base de datos tienda, muestra la descripción de las tablas articulo, puesto y venta.**

```
use tienda;
show databases;
show tables;
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	

```
use tienda;
show databases;
show tables;
describe puesto;
```

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

```
use tienda;
show databases;
show tables;
describe venta;
```

	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	timestamp	NO		CURRENT_TIMESTAMP	DEFAULT_GENERA

## RETO 2

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

```
select * from empleado where id_puesto =4;
```

	id_empleado	id_puesto	nombre	apellido_paterno	apellido_materno	rfc
▶	317	4	Norrie	McGarrie	Hartopp	HDOW177256O16
	361	4	Maxy	Udden	Kose	MBZF302490Y06
▲	NULL	NULL	NULL	NULL	NULL	NULL

**02. ¿Qué puestos tienen un salario mayor a \$10,000?**

```
select * from puesto where salario > 1000;
```

	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

**03. ¿Qué artículos tienen un precio mayor a \$1,000 y un iva mayor a 100?**

```
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

**04. ¿Qué ventas incluyen los artículo 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);

	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

## RETO 3

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

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

## PROYECTO

**01. -- Dentro del mismo servidor de bases de datos, conéctate al esquema classicmodels.**

use classicmodels;

**02. -- Dentro de la tabla employees, obtén el apellido de todos los empleados.**

select lastName from employees;

	lastName
▶	Murphy
	Patterson
	Firrelli
	Patterson
	Bondur
	Bow
	Jennings

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

select lastName, firstName, jobTitle from employees;

	lastName	firstName	jobTitle
▶	Murphy	Diane	President
	Patterson	Mary	VP Sales
	Firrelli	Jeff	VP Marketing
	Patterson	William	Sales Manager (APAC)
	Bondur	Gerard	Sale Manager (EMEA)
	Bow	Anthony	Sales Manager (NA)
	Jennings	Leslie	Sales Rep

**04. -- Dentro de la tabla employees, obtén todos los datos de cada empleado.**

select \* from employees;

	employeeNumber	lastName	firstName	extension	email	officeCode	
▶	1002	Murphy	Diane	x5800	dmurphy@classicmodelcars.com	1	1
	1056	Patterson	Mary	x4611	mpatterso@classicmodelcars.com	1	1
	1076	Firrelli	Jeff	x9273	jfirrelli@classicmodelcars.com	1	1
	1088	Patterson	William	x4871	wpatterson@classicmodelcars.com	6	1
	1102	Bondur	Gerard	x5408	gbondur@classicmodelcars.com	4	1
	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1
	1165	Jennings	Leslie	x3291	ljennings@classicmodelcars.com	1	1
	1166	Thompson	Leslie	x4065	lthompson@classicmodelcars.com	1	1

**05. -- 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";

	lastName	firstName	jobTitle
▶	Jennings	Leslie	Sales Rep
	Thompson	Leslie	Sales Rep
	Firrelli	Julie	Sales Rep
	Patterson	Steve	Sales Rep
	Tseng	Foon Yue	Sales Rep
	Vanauf	George	Sales Rep
	Bondur	Loui	Sales Rep

**06. -- 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;

	lastName	firstName	jobTitle	officeCode
▶	Jennings	Leslie	Sales Rep	1
	Thompson	Leslie	Sales Rep	1

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

select lastName, firstName, jobTitle, officeCode from employees where jobTitle = "Sales Rep" or officeCode = 1;

	lastName	firstName	jobTitle	officeCode
▶	Murphy	Diane	President	1
	Patterson	Mary	VP Sales	1
	Firrelli	Jeff	VP Marketing	1
	Bow	Anthony	Sales Manager (NA)	1
	Jennings	Leslie	Sales Rep	1
	Thompson	Leslie	Sales Rep	1
	Firrelli	Julie	Sales Rep	2

**08. -- 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);

	lastName	firstName	officeCode
▶	Murphy	Diane	1
	Patterson	Mary	1
	Firrelli	Jeff	1
	Bow	Anthony	1
	Jennings	Leslie	1
	Thompson	Leslie	1
	Firrelli	Julie	2

**09. -- 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";

	lastName	firstName	jobTitle
▶	Murphy	Diane	President
	Patterson	Mary	VP Sales
	Firrelli	Jeff	VP Marketing
	Patterson	William	Sales Manager (APAC)
	Bondur	Gerard	Sale Manager (EMEA)
	Bow	Anthony	Sales Manager (NA)

- 10. -- 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;

	lastName	firstName	officeCode
▶	Patterson	William	6
	Bott	Larry	7
	Jones	Barry	7
	Fixter	Andy	6
	Marsh	Peter	6
	King	Tom	6

- 11. -- Dentro de la tabla employees, obtén el apellido, nombre y código de oficina de todos los empleados cuyo código de oficina sea menor o igual a 4.**

select lastName, firstName, officeCode from employees where officeCode <= 4;

	lastName	firstName	officeCode
▶	Murphy	Diane	1
	Patterson	Mary	1
	Firrelli	Jeff	1
	Bondur	Gerard	4
	Bow	Anthony	1
	Jennings	Leslie	1
	Thompson	Leslie	1

- 12. -- 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";

	customerName	country	state
▶	Mini Gifts Distributors Ltd.	USA	CA
	Mini Wheels Co.	USA	CA
	Technics Stores Inc.	USA	CA
	Toys4GrownUps.com	USA	CA
	Boards & Toys Co.	USA	CA
	Collectable Mini Designs Co.	USA	CA
	Corporate Gift Ideas Co.	USA	CA

- 13. -- 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;

	customerName	country	state	creditLimit
▶	Mini Gifts Distributors Ltd.	USA	CA	210500.00
	Collectable Mini Designs Co.	USA	CA	105000.00
	Corporate Gift Ideas Co.	USA	CA	105000.00

- 14. -- 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");

	customerName	country
▶	Atelier graphique	France
	Signal Gift Stores	USA
	La Rochelle Gifts	France
	Mini Gifts Distributors Ltd.	USA
	Mini Wheels Co.	USA
	Land of Toys Inc.	USA
	Boards & Toys Co.	France

- 15. -- 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;

	customerName	country	creditLimit
▶	La Rochelle Gifts	France	118200.00
	Mini Gifts Distributors Ltd.	USA	210500.00
	Land of Toys Inc.	USA	114900.00
	Saveley & Henriot, Co.	France	123900.00
	Muscle Machine Inc	USA	138500.00
	Diecast Classics Inc.	USA	100600.00
	Collectable Mini Designs Co.	USA	105000.00

- 16. -- 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 where country in ("USA","France");

	officeCode	city	phone	country
▶	1	San Francisco	+1 650 219 4782	USA
	2	Boston	+1 215 837 0825	USA
	3	NYC	+1 212 555 3000	USA
	4	Paris	+33 14 723 4404	France
*	NULL	NULL	NULL	NULL

- 17. -- 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";

	officeCode	city	phone	country
▶	5	Tokyo	+81 33 224 5000	Japan
	6	Sydney	+61 2 9264 2451	Australia
	7	London	+44 20 7877 2041	UK
*	NULL	NULL	NULL	NULL

- 18. -- 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 where orderNumber in (10165,10287,10310);

	orderNumber	customerNumber	status	shippedDate
▶	10165	148	Shipped	2003-12-26
	10287	298	Shipped	2004-09-01
	10310	259	Shipped	2004-10-18
*	NULL	NULL	NULL	NULL

- 19. -- 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;

	contactLastName	contactFirstName
▶	Accorti	Paolo
	Altagar, G M	Raanan
	Andersen	Mel
	Anton	Carmen
	Ashworth	Rachel
	Barajas	Miguel
	Benitez	Violeta

- 20. -- 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;

	contactLastName	contactFirstName
▶	Young	Jeff
	Young	Julie
	Young	Mary
	Young	Dorothy
	Yoshido	Juri
	Walker	Brydey
	Victorino	Wendy

- 21. -- 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;

	contactLastName	contactFirstName
▶	Young	Dorothy
	Young	Jeff
	Young	Julie
	Young	Mary
	Yoshido	Juri
	Walker	Brydey
	Victorino	Wendy

- 22. -- 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;

	customerNumber	customerName	creditLimit
▶	141	Euro+ Shopping Channel	227600.00
	124	Mini Gifts Distributors Ltd.	210500.00
	298	Vida Sport, Ltd	141300.00
	151	Muscle Machine Inc	138500.00
	187	AV Stores, Co.	136800.00
*	NULL	NULL	NULL

- 23. -- 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 ASC limit 5;

	customerNumber	customerName	creditLimit
▶	223	Natürlich Autos	0.00
	168	American Souvenirs Inc	0.00
	169	Porto Imports Co.	0.00
	206	Asian Shopping Network, Co	0.00
	125	Havel & Zbyszek Co	0.00
*	NULL	NULL	NULL

## SESIÓN 02

### RETO 1

Usando la base de datos *tienda*, escribe consultas que permitan responder las siguientes preguntas.

**01. ¿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	6537.91	1113.99	906
	213	Pasta - Rotini, Colour, Dry	1830.13	373.98	309
	233	Pasta - Cannelloni, Sheets, Fresh	2316.37	605.55	307

**02. ¿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

**03. ¿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
	6	Yeast Dry - Fleischman	923.18	524.08	818
	7	Nantucket - Kiwi Berry Ckrl.	5579.47	1012.33	527

### RETO 2

Usando la base de datos *tienda*, escribe consultas que permitan responder las siguientes preguntas.

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

```
select avg(salario) from puesto;
```

	avg(salario)
▶	19595.051179999973

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

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

	count(*)
▶	17

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

```
select min(salario), max(salario) from puesto;
```

	min(salario)	max(salario)
▶	10013.44	29996.58

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

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

	sum(salario)
▶	98919.69



## RETO 3

Usando la base de datos *tienda*, escribe consultas que permitan responder las siguientes preguntas.

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

```
select nombre, count(id_puesto) from puesto group by nombre;
```

nombre	count(id_puesto)
Analog Circuit Design manager	8
Junior Executive	8
Director of Sales	8
Staff Scientist	9
Desktop Support Technician	5
Budget/Accounting Analyst III	4
Accounting Assistant III	3

**02. ¿Cuánto dinero se paga en total por puesto?**

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

nombre	sum(salario)
Analog Circuit Design manager	179310.18000000002
Junior Executive	156846.26
Director of Sales	136630.69
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

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

```
SELECT id_empleado, COUNT(id_venta) FROM venta GROUP BY id_empleado;
```

id_empleado	count(id_venta)
2	2
3	2
4	1
5	1
6	2
12	5
15	4
18	2
22	1

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

```
SELECT id_articulo, COUNT(*) FROM venta GROUP BY id_articulo;
```

id_articulo	count(id_venta)
2	1
3	1
4	2
8	1
10	1
11	1
12	1
13	2
15	1

## RETO 4

Usando la base de datos *tienda*, escribe consultas que permitan responder las siguientes preguntas.

**01. ¿Cuál es el nombre de los empleados cuyo sueldo es menor a \$20,000?**

```
SELECT nombre, apellido_paterno FROM empleado WHERE id_puesto IN  
(SELECT id_puesto FROM puesto WHERE salario < "20000");
```



	nombre	apellido_paterno
►	Norrie	McGarrie
	Maxy	Udden
	Della	Fulbrook
	Katya	Banbridge
	Robyn	Hancock
	Hayyim	Verdon
	Analise	Beteriss
	Artair	Dearn
	Penny	Dargie
	Doy	Pepperrall
	Orly	Punch
	Rockwell	Euels
	Antonin	Oneill
	Mozes	Proudlock
	Emmie	Pamphilon
	Dulciana	Livick
	Jacquelin	Crystal
	Fair	Minker
	Louie	Okker
	Debee	Kingshott
	Berna...	Petroselli
	Edee	Billin

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

```
SELECT id_empleado,min(ventas),max(ventas) FROM
(select clave,id_empleado,COUNT(*) AS ventas FROM venta GROUP BY clave,id_empleado) AS vent
GROUP BY id_empleado;
```

	id_empleado	min(ventas)	max(ventas)
►	2	1	1
	3	1	1
	4	1	1
	5	1	1
	6	1	1
	12	1	1
	15	1	2
	18	1	1
	22	1	1
	23	1	1
	25	1	1
	26	1	1

**03. ¿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)
FROM empleado AS e;
```

	nombre	apellido_paterno	(SELECT nombre FROM puesto p WHERE p.id_puesto = e.id_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
	Andre	Theurer	Tax Accountant
	Land	Locksley	Product Engineer
	Nikki	Fayerbrother	Sales Associate
	Aldridge	Bidewell	Structural Engineer
	Jerald	Warboy	GIS Technical Architect
	Crissie	Cleveley	Help Desk Technician
	Cristian	Fisby	Business Systems Development Analyst

## PROYECTO

- 01. 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%";

	employeeNumber	lastName	firstName
▶	1143	Bow	Anthony
	1611	Fixter	Andy
✱	NULL	NULL	NULL

- 02. 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";

	employeeNumber	lastName	firstName
✱	NULL	NULL	NULL

- 03. 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
▶	1143	Bow	Anthony
	1286	Tseng	Foon Yue
✱	NULL	NULL	NULL

- 04. 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.**

- 05.** SELECT employeeNumber, lastName, firstName FROM employees WHERE firstName LIKE "t\_m";

	employeeNumber	lastName	firstName
▶	1619	King	Tom
✱	NULL	NULL	NULL

- 06. 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
▶	1002	Murphy	Diane
	1056	Patterson	Mary
	1076	Firrelli	Jeff
	1088	Patterson	William
	1102	Bondur	Gerard
	1143	Bow	Anthony
	1165	Jennings	Leslie
	1166	Thompson	Leslie
	1188	Firrelli	Julie
	1216	Patterson	Steve
	1286	Tseng	Foon Yue
	1323	Vanauf	George
	1337	Bondur	Loui
	1370	Hernandez	Gerard

- 07. 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

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

SELECT orderNumber,COUNT(\*) FROM orderdetails GROUP BY orderNumber;

	orderNumber	COUNT(*)
▶	10100	4
	10101	4
	10102	2
	10103	16
	10104	13
	10105	15
	10106	18
	10107	8
	10108	16
	10109	6
	10110	16
	10111	6
	10112	2
	10113	4
	10114	10
	10115	5
	10116	1
	10117	12

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

SELECT YEAR(orderDate) AS Año,COUNT(\*) AS ordenes FROM orders GROUP BY year(orderDate);

	Año	ordenes
▶	2003	111
	2004	151
	2005	64

**10. 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");

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

**11. 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 ORDER BY amount DESC LIMIT 1;

	customerNumber	checkNumber	amount
▶	141	JE105477	120166.58
*	NULL	NULL	NULL

**12. 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);

	customerNumber	checkNumber	amount
	276	LE432182	41554.73
	278	BJ483870	37654.09
	278	GP636783	52151.81
	278	NI983021	37723.79
	282	JT819493	35806.73
	286	DR578578	47411.33
	286	KH910279	43134.04
	298	AJ574927	47375.92
	298	LF501133	61402.00
	299	AD304085	36798.88
	311	DG336041	46770.52
	311	FA728475	32723.04
	314	LQ244073	45352.47
	319	HL685576	42339.76
	319	OM548174	36092.40
	320	HO576374	41016.75
	320	MU817160	52548.49
	321	DJ15149	85559.12
	321	LA556321	46781.66
	323	AL493079	75020.13
	323	ES347491	37281.36
	323	PQ803830	39440.59
	324	HR150714	37455.77

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

SELECT customerName FROM customers WHERE customerNumber NOT IN (SELECT customerNumber FROM orders);

customerName
Havel & Zbyszek Co
American Souvenirs Inc
Porto Imports Co.
Asian Shopping Network, Co
Natürlich Autos
ANG Resellers
Messner Shopping Network
Franken Gifts, Co
BG&E Collectables
Schuyler Imports
Der Hund Imports
Cramer Spezialitäten, Ltd
Asian Treasures, Inc.
SAR Distributors, Co
Kommission Auto
Lisboa Souvenirs, Inc
Precious Collectables
Stuttgart Collectable Excha...
Feuer Online Stores, Inc
Warburg Exchange
Anton Designs, Ltd.
Mit Vergnügen & Co.
Kremlin Collectables, Co.

**14. 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

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

SELECT status,COUNT(\*) FROM orders GROUP BY status;

	status	COUNT(*)
►	Shipped	303
	Resolved	4
	Cancelled	6
	On Hold	4
	Disputed	3
	In Process	6

## SESIÓN 03

### RETO 1

Usando la base de datos *tienda*, escribe consultas que permitan responder las siguientes preguntas.

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

```
SELECT clave,nombre,apellido_paterno FROM venta AS v  
JOIN empleado AS e  
ON v.id_empleado = e.id_empleado  
ORDER BY clave;
```

	clave	nombre	apellido_paterno
▶	0002-8149	Leonidas	Junkinson
	0002-8149	Leslie	Cuvley
	0002-8149	Edee	Billin
	0002-8149	Jillie	Corter
	0002-8149	Sydney	Woolway
	0002-8149	Aguistin	Richarz
	0002-8149	Sigfrid	Teal
	0002-8149	Rebecka	Rushworth
	0002-8149	Nora	O'Suaird
	0002-8149	Luise	Lennard
	0002-8149	Gustie	Gryglewski
	0002-8149	Petr	Skedgell
	0002-8149	Doti	Mc Kellen
	0008-0833	Kain	Glassopp

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

```
SELECT clave, nombre FROM venta AS v  
JOIN articulo AS a  
ON v.id_articulo = a.id_articulo  
ORDER BY clave;
```

	clave	nombre
▶	0002-8149	Sprouts - Pea
	0002-8149	Wine - Saint - Bris 2002, Sauv
	0002-8149	Sauce Tomato Pouch
	0002-8149	Wine - Montecillo Rioja Crianza
	0002-8149	Yokaline
	0002-8149	Marsala - Sperone, Fine, D.o.c.
	0002-8149	Glaze - Clear
	0002-8149	Beans - Green
	0002-8149	Water, Tap
	0002-8149	Butter Sweet
	0002-8149	Sour Puss - Tangerine
	0002-8149	Bread - White, Sliced
	0002-8149	Juice - Orange 1.89l
	0008-0833	Ice Cream Bar - Drumstick

**03. ¿Cuál es el total de cada venta?**

```
SELECT clave,sum(precio) FROM venta AS v  
JOIN articulo AS a  
ON v.id_articulo = a.id_articulo  
GROUP BY clave  
ORDER BY clave;
```

	dave	sum(precio)
►	0002-8149	38642.55000000001
	0008-0833	862.6
	0009-0347	6879.56
	0019-9888	178.57
	0023-9177	3891.89
	0049-0032	321524.60999999999
	0065-0396	5571.99
	0067-5080	1675.53
	0067-6122	87221.89
	0069-1001	5382.4
	0074-6633	2882.74
	0075-2915	2799.84
	0078-0326	1652.68
	0078-0446	1157.47

## RETO 2

Usando la base de datos *tienda*, define las siguientes vistas que permitan obtener la siguiente información.

### 01. Obtener el puesto de un empleado.

```
CREATE VIEW ycpuesto AS
(SELECT CONCAT(e.nombre," ",e.apellido_paterno) AS empleado,p.nombre
FROM empleado AS e
JOIN puesto AS p
ON e.id_puesto = p.id_puesto);
SELECT * FROM ycpuesto;
```

	empleado	nombre
►	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
	Andre Theurer	Tax Accountant
	Land Locksley	Product Engineer
	Nikki Fayerbrother	Sales Associate
	Aldridge Bidewell	Structural Engineer
	Jerald Warboy	GIS Technical Architect
	Crissie Cleveley	Help Desk Technician
	Cristian Fisby	Business Systems Developme...
	Erskine Oxenden	Data Coordinator

### 02. Saber qué artículos ha vendido cada empleado.

```
CREATE VIEW ycmpleado AS
(SELECT CONCAT(e.nombre," ",e.apellido_paterno) AS empleado,a.nombre
FROM venta AS v
JOIN empleado AS e
ON v.id_empleado = e.id_empleado
JOIN articulo AS a
ON v.id_articulo = a.id_articulo);
```



empleado	nombre
Arlana Fanstone	Cake - Cake Sheet Macaroon
Juliana Gecke	Beans - Long, Chinese
Maryellen Parkisson	Beef - Rib Roast, Capless
Cad Sambedge	Langers - Ruby Red Grapfruit
Michaeline Stoli	Filter - Coffee
Courtney Harston	Corn Meal
Giuditta Chicchetto	Wine - White, Pelee Island
Juliana Gecke	Bagelers - Cinn / Brown
Rosalinde Livock	Smoked Tongue
Ema Itzkovich	Sage Ground Wiberg
Emmy Roblin	Duck - Breast
Mignonne Baszniak	Filter - Coffee
Hersch Blader	Strawberries - California
Hyacinthia Petz	Cake - Box Window 10x10x2.5

### 03. Saber qué puesto ha tenido más ventas.

```
CREATE VIEW ycventa AS
(SELECT p.nombre,COUNT(v.id_venta) AS ventas FROM venta AS v
JOIN empleado AS e
ON v.id_empleado = e.id_empleado
JOIN puesto AS p
ON p.id_puesto = e.id_puesto
GROUP BY p.nombre
ORDER BY ventas DESC
LIMIT 1);
```

nombre	ventas
Physical Therapy Assistant	23

## PROYECTO

Para estas consultas usa **RIGHT JOIN**

### 01. Obten el código de producto, nombre de producto y descripción de todos los productos.

```
SELECT productCode,productName,productDescription FROM products;
```

productCode	productName	productDescription
S10_1678	1969 Harley Davidson Ultimate Chopper	This replica features working kickstand, front su...
S10_1949	1952 Alpine Renault 1300	Turnable front wheels; steering function; detail...
S10_2016	1996 Moto Guzzi 1100i	Official Moto Guzzi logos and insignias, saddle b...
S10_4698	2003 Harley-Davidson Eagle Drag Bike	Model features, official Harley Davidson logos a...
S10_4757	1972 Alfa Romeo GTA	Features include: Turnable front wheels; steeri...
S10_4962	1962 LanciaA Delta 16V	Features include: Turnable front wheels; steeri...
S12_1099	1968 Ford Mustang	Hood, doors and trunk all open to reveal highly ...
S12_1108	2001 Ferrari Enzo	Turnable front wheels; steering function; detail...
S12_1666	1958 Setra Bus	Model features 30 windows, skylights & glare re...
S12_2823	2002 Suzuki XREO	Official logos and insignias, saddle bags located ...
S12_3148	1969 Corvair Monza	1:18 scale die-cast about 10" long doors open, ...
S12_3380	1968 Dodge Charger	1:12 scale model of a 1968 Dodge Charger. Ho...
S12_3891	1969 Ford Falcon	Turnable front wheels; steering function; detail...
S12_3990	1970 Plymouth Hemi Cuda	Very detailed 1970 Plymouth Cuda model in 1:1...

### 02. Obten el número de orden, estado y costo total de cada orden.

```
SELECT o.orderNumber,o.status,SUM(od.priceEach) AS costototal
FROM orders AS o
RIGHT JOIN orderdetails AS od
ON o.orderNumber = od.orderNumber
GROUP BY o.orderNumber;
```

	orderNumber	status	costototal
▶	10100	Shipped	301.84
	10101	Shipped	352.00
	10102	Shipped	138.68
	10103	Shipped	1520.37
	10104	Shipped	1251.89
	10105	Shipped	1479.71
	10106	Shipped	1427.28
	10107	Shipped	793.21
	10108	Shipped	1432.86
	10109	Shipped	700.89
	10110	Shipped	1338.47
	10111	Shipped	460.16
	10112	Shipped	282.26
	10113	Shipped	325.23

**03. 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,od.orderLineNumber,p.productName,od.quantityOrdered,od.priceEach

FROM orders AS o

RIGHT JOIN orderdetails AS od

ON o.orderNumber = od.orderNumber

RIGHT JOIN products AS p

ON od.productCode = p.productCode

ORDER BY o.orderNumber;

	orderNumber	orderDate	orderLineNumber	productName	quantityOrdered	priceEach
▶	NULL	NULL	NULL	1985 Toyota Supra	NULL	NULL
	10100	2003-01-06	3	1917 Grand Touring Sedan	30	136.00
	10100	2003-01-06	2	1911 Ford Town Car	50	55.09
	10100	2003-01-06	4	1932 Alfa Romeo 8C2300 Spider Sport	22	75.46
	10100	2003-01-06	1	1936 Mercedes Benz 500k Roadster	49	35.29
	10101	2003-01-09	4	1932 Model A Ford J-Coupe	25	108.06
	10101	2003-01-09	1	1928 Mercedes-Benz SSK	26	167.06
	10101	2003-01-09	3	1939 Chevrolet Deluxe Coupe	45	32.53
	10101	2003-01-09	2	1938 Cadillac V-16 Presidential Limousine	46	44.35
	10102	2003-01-10	2	1937 Lincoln Berline	39	95.55
	10102	2003-01-10	1	1936 Mercedes-Benz 500K Special Road...	41	43.13
	10103	2003-01-29	11	1952 Alpine Renault 1300	26	214.30
	10103	2003-01-29	4	1962 LanciaA Delta 16V	42	119.67
	10103	2003-01-29	8	1958 Setra Bus	27	121.64

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

SELECT od.orderNumber,p.productName,p.MSRP,od.priceEach FROM orderdetails AS od

JOIN products AS p

ON od.productCode = p.productCode

ORDER BY od.orderNumber;

	orderNumber	productName	MSRP	priceEach
▶	10100	1917 Grand Touring Sedan	170.00	136.00
	10100	1911 Ford Town Car	60.54	55.09
	10100	1932 Alfa Romeo 8C2300 Spider Sport	92.03	75.46
	10100	1936 Mercedes Benz 500k Roadster	41.03	35.29
	10101	1932 Model A Ford J-Coupe	127.13	108.06
	10101	1928 Mercedes-Benz SSK	168.75	167.06
	10101	1939 Chevrolet Deluxe Coupe	33.19	32.53
	10101	1938 Cadillac V-16 Presidential Limousine	44.80	44.35
	10102	1937 Lincoln Berline	102.74	95.55
	10102	1936 Mercedes-Benz 500K Special Roadster	53.91	43.13
	10103	1952 Alpine Renault 1300	214.30	214.30
	10103	1962 LanciaA Delta 16V	147.74	119.67
	10103	1958 Setra Bus	136.67	121.64
	10103	1940 Ford Pickup Truck	116.67	94.50

Para estas consultas usa **LEFT JOIN**

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

```
SELECT c.customerNumber,c.customerName,o.orderNumber,c.state FROM customers AS c
LEFT JOIN orders AS o
ON c.customerNumber = o.customerNumber
ORDER BY c.customerNumber;
```

	customerNumber	customerName	orderNumber	state
▶	103	Atelier graphique	10123	NULL
	103	Atelier graphique	10298	NULL
	103	Atelier graphique	10345	NULL
	112	Signal Gift Stores	10124	NV
	112	Signal Gift Stores	10278	NV
	112	Signal Gift Stores	10346	NV
	114	Australian Collectors, Co.	10120	Victoria
	114	Australian Collectors, Co.	10125	Victoria
	114	Australian Collectors, Co.	10223	Victoria
	114	Australian Collectors, Co.	10342	Victoria
	114	Australian Collectors, Co.	10347	Victoria
	119	La Rochelle Gifts	10275	NULL
	119	La Rochelle Gifts	10315	NULL
	119	La Rochelle Gifts	10375	NULL

**06. Obtén los clientes que no tienen una orden asociada.**

```
SELECT c.customerName FROM customers AS c
LEFT JOIN orders AS o
ON c.customerNumber != o.customerNumber;
```

	customerName
▶	Signal Gift Stores
	Australian Collectors, Co.
	La Rochelle Gifts
	Baane Mini Imports
	Mini Gifts Distributors Ltd.
	Havel & Zbyszek Co
	Blauer See Auto, Co.
	Mini Wheels Co.
	Land of Toys Inc.
	Euro+ Shopping Channel
	Volvo Model Replicas, Co
	Danish Wholesale Imports
	Saveley & Henriot, Co.
	Dragon Souvenirs, Ltd.

**07. 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.customerName,py.checkNumber,py.amount AS total
FROM employees AS e
```

```

LEFT JOIN customers AS c
ON e.employeeNumber = c.salesRepEmployeeNumber
LEFT JOIN payments AS py
ON c.customerNumber = py.customerNumber
ORDER BY e.lastName;

```

	lastName	firstName	customerName	checkNumber	total
▶	Bondur	Gerard	NULL	NULL	NULL
	Bondur	Loui	Saveley & Henriot, Co.	FP549817	40978.53
	Bondur	Loui	Saveley & Henriot, Co.	FU793410	49614.72
	Bondur	Loui	Saveley & Henriot, Co.	LJ160635	39712.10
	Bondur	Loui	La Corne D'abondance, Co.	AD832091	1960.80
	Bondur	Loui	La Corne D'abondance, Co.	CE51751	51209.58
	Bondur	Loui	La Corne D'abondance, Co.	EH208589	33383.14
	Bondur	Loui	Lyon Souvenirs	EQ12267	17928.09
	Bondur	Loui	Lyon Souvenirs	HD284647	26311.63
	Bondur	Loui	Lyon Souvenirs	HN114306	23419.47
	Bondur	Loui	Marseille Mini Autos	BQ602907	18888.31
	Bondur	Loui	Marseille Mini Autos	CI471510	50824.66
	Bondur	Loui	Marseille Mini Autos	OB648482	1834.56
	Bondur	Loui	Reims Collectables	CO351193	49705.52

**Para estas consultas usa RIGHT JOIN**

**08. Repite los ejercicios 5 a 7 usando RIGHT JOIN.**

```

5. SELECT c.customerNumber,c.customerName,o.orderNumber,c.state
FROM customers AS c
RIGHT JOIN orders AS o
ON c.customerNumber = o.customerNumber
ORDER BY c.customerNumber;

```

	customerNumber	customerName	orderNumber	state
▶	103	Atelier graphique	10123	NULL
	103	Atelier graphique	10298	NULL
	103	Atelier graphique	10345	NULL
	112	Signal Gift Stores	10124	NV
	112	Signal Gift Stores	10278	NV
	112	Signal Gift Stores	10346	NV
	114	Australian Collectors, Co.	10120	Victoria
	114	Australian Collectors, Co.	10125	Victoria
	114	Australian Collectors, Co.	10223	Victoria
	114	Australian Collectors, Co.	10342	Victoria
	114	Australian Collectors, Co.	10347	Victoria
	119	La Rochelle Gifts	10275	NULL
	119	La Rochelle Gifts	10315	NULL
	119	La Rochelle Gifts	10375	NULL

```

6. SELECT c.customerName FROM orders AS o
RIGHT JOIN customers AS c
ON o.customerNumber != c.customerName;

```

	customerName
▶	Atelier graphique
	Signal Gift Stores
	Australian Collectors, Co.
	La Rochelle Gifts
	Baane Mini Imports
	Mini Gifts Distributors Ltd.
	Havel & Zbyszek Co
	Blauer See Auto, Co.
	Mini Wheels Co.
	Land of Toys Inc.
	Euro+ Shopping Channel
	Volvo Model Replicas, Co
	Danish Wholesale Imports
	Saveley & Henriot, Co.

```

7. SELECT e.lastName,e.firstName,c.customerName,py.checkNumber,py.amount AS total
FROM payments AS py
RIGHT JOIN customers AS c
ON py.customerNumber = c.customerNumber
RIGHT JOIN employees AS e
ON c.salesRepEmployeeNumber = e.employeeNumber
ORDER BY e.lastName;

```


	lastName	firstName	customerName	checkNumber	total
▶	Bondur	Gerard	NULL	NULL	NULL
	Bondur	Loui	Saveley & Henriot, Co.	FP549817	40978.53
	Bondur	Loui	Saveley & Henriot, Co.	FU793410	49614.72
	Bondur	Loui	Saveley & Henriot, Co.	LJ160635	39712.10
	Bondur	Loui	La Corne D'abondance, Co.	AD832091	1960.80
	Bondur	Loui	La Corne D'abondance, Co.	CE51751	51209.58
	Bondur	Loui	La Corne D'abondance, Co.	EH208589	33383.14
	Bondur	Loui	Lyon Souvenirs	EQ12267	17928.09
	Bondur	Loui	Lyon Souvenirs	HD284647	26311.63
	Bondur	Loui	Lyon Souvenirs	HN114306	23419.47
	Bondur	Loui	Marseille Mini Autos	BQ602907	18888.31
	Bondur	Loui	Marseille Mini Autos	CI471510	50824.66
	Bondur	Loui	Marseille Mini Autos	OB648482	1834.56
	Bondur	Loui	Reims Collectables	CO351193	49705.52

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

```

CREATE VIEW ycestatus AS
(SELECT o.orderNumber,o.status,SUM(od.priceEach) AS costototal
FROM orders AS o
RIGHT JOIN orderdetails AS od
ON o.orderNumber = od.orderNumber
GROUP BY o.orderNumber);

```


▼  ycestatus

- ◆ orderNumber
- ◆ status
- ◆ costototal

```

CREATE VIEW ycprecios AS
(SELECT od.orderNumber,p.productName,p.MSRP,od.priceEach FROM orderdetails AS od
JOIN products AS p
ON od.productCode = p.productCode
ORDER BY od.orderNumber);

```


▼  ycprecios

- ◆ orderNumber
- ◆ productName
- ◆ MSRP
- ◆ priceEach

```

CREATE VIEW ycsalesrep AS
(SELECT e.lastName,e.firstName,c.customerName,py.checkNumber,py.amount AS total
FROM payments AS py
RIGHT JOIN customers AS c
ON py.customerNumber = c.customerNumber
RIGHT JOIN employees AS e
ON c.salesRepEmployeeNumber = e.employeeNumber
ORDER BY e.lastName);

```

▼  ycsalesrep

- ◆ lastName
- ◆ firstName
- ◆ customerName
- ◆ checkNumber
- ◆ total



## SESIÓN 04

### RETO 1

Usando la base de datos *sample\_mflix*, proyecta los datos que se solicitan.

#### 01. 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
```

---

```
name: "Talisa Maegyr"  
text: "Rem itaque ad sit rem voluptatibus. Ad fugiat maxime illum optio iure ..."  
date: 1998-08-22T11:45:03.000+00:00
```

---

```
name: "Cameron Duran"  
text: "Quasi dicta culpa asperiores quaerat perferendis neque. Est animi pari..."  
date: 1983-04-27T20:39:15.000+00:00
```

---

#### 02. Título, elenco y año de cada película.

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

```
▶ cast: Array  
  title: "Blacksmith Scene"  
  year: 1893
```

---

```
▶ cast: Array  
  title: "The Great Train Robbery"  
  year: 1903
```

---

```
▶ cast: Array  
  title: "The Land Beyond the Sunset"  
  year: 1912
```

---

```
▶ cast: Array  
  title: "A Corner in Wheat"  
  year: 1909
```

---

#### 03. Nombre y contraseña de cada usuario.

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



```
name: "Ned Stark"
password: "$2b$12$UREFwsRUoyF0CRqGNK0Lz00HM/jLhgUCNNIJ9RJAqMUQ74cr1J1Vu"
```

```
name: "Robert Baratheon"
password: "$2b$12$yGqxLG9LZpXA2xVDhuPnSOZd.VURVxz7wgOLY3pn00s7u2S1Z032y"
```

```
name: "Jaime Lannister"
password: "$2b$12$6vz7wiwO.EI5Rilvq1zUc./9480gb1uPtXcahDxIadgyC3PS8XCUK"
```

```
name: "Catelyn Stark"
password: "$2b$12$fiaTH5Sh1zKNFX2i/FTereWGjxoJxvmV7XL.q1fqCr8CwOxK.mZWS"
```

## RETO 2

Usando la base de datos *sample\_mflix*, agrega proyecciones, filtros, ordenamientos y límites que permitan contestar las siguientes preguntas.

### 01. ¿Qué comentarios ha hecho Greg Powell?

```
{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
```

```
_id: ObjectId("5a9427648b0beeb6957afe")
name: "Greg Powell"
email: "greg_powell@fakegmail.com"
movie_id: ObjectId("573a1391f29313caabcd754b")
text: "Rem nostrum nobis saepe eaque itaque nemo. Fugit dignissimos nisi sapi..."
date: 2013-03-26T16:20:03.000+00:00
```

```
_id: ObjectId("5a9427648b0beeb6957b56")
name: "Greg Powell"
email: "greg_powell@fakegmail.com"
movie_id: ObjectId("573a1391f29313caabcd7e5d")
text: "Officia atque ullam esse doloribus laborum. Maiores dicta ratione rem ..."
date: 2004-04-08T08:21:05.000+00:00
```

### 02. ¿Qué comentarios han hecho Greg Powell o Mercedes Tyler?

```
{$or:[{name:"Greg Powell"},{name:"Mercedes Tyler"}]}
```

```
_id: ObjectId("5a9427648b0beebeb69579cf")
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
```

```
_id: ObjectId("5a9427648b0beebeb69579e7")
name: "Mercedes Tyler"
email: "mercedes_tyler@fakegmail.com"
movie_id: ObjectId("573a1390f29313caabcd4323")
text: "Eius veritatis vero facilis quaerat fuga temporibus. Praesentium exped..."
date: 2002-08-18T04:56:07.000+00:00
```

```
_id: ObjectId("5a9427648b0beebeb6957a78")
name: "Mercedes Tyler"
email: "mercedes_tyler@fakegmail.com"
movie_id: ObjectId("573a1390f29313caabcd6399")
text: "Voluptate odio minima pariatur recusandae. Architecto illum dicta repu..."
date: 2007-10-17T06:50:56.000+00:00
```

### 03. ¿Cuál es el máximo número de comentarios en una película?

```
{num_mflix_comments: 1}
{num_mflix_comments: -1}
Limit: 1
```

```
_id: ObjectId("573a1399f29313caabcee886")
num_mflix_comments: 456
```

### 04. ¿Cuál es título de las cinco películas más comentadas?

```
{title: 1}
{num_mflix_comments: -1}
Limit: 5
```

```
_id: ObjectId("573a1399f29313caabcee886")
title: "The Mask"
```

```
_id: ObjectId("573a1399f29313caabcee578")
title: "Dumb & Dumber"
```

```
_id: ObjectId("573a13bff29313caabd6001f")
title: "The Unborn"
```

```
_id: ObjectId("573a13a5f29313caabd159a9")
title: "About a Boy"
```

```
_id: ObjectId("573a13a7f29313caabd1aa55")
title: "8 Mile"
```

## PROYECTO

Las consultas se realizarán sobre la base `sample_training`.

Todas las consultas que realices deberás mantenerlas dentro del MongoDB Compass. Para hacer esto, da clic en el botón con los puntos `...` y en *Toogle Query History*. Busca la última consulta y agregala a favoritos presionando el ícono con la estrella ☆

### 01. Obtén los datos de contacto de cada compañía.

```
{
  project: {
    name: 1,
    email_address: 1,
    phone_number: 1
  }
}
```

```

}
PROJECT {name:1,email_address:1,phone_number:1}
SORT
COLLATION
SKIP 0
LIMIT 0
VIEW [ ] [ ] [ ]
Displaying documents 1 - 20 of 9500

_id: ObjectId("52cdef7c4bab8bd675297d8a")
name: "Wetpaint"
email_address: "info@wetpaint.com"
phone_number: "206.859.6300"

_id: ObjectId("52cdef7c4bab8bd675297d8e")
name: "Facebook"
email_address: ""
phone_number: ""

_id: ObjectId("52cdef7c4bab8bd675297d8f")
name: "Omnidrive"
email_address: "info@omnidrive.com"
phone_number: "660-675-5052"

_id: ObjectId("52cdef7c4bab8bd675297d94")
name: "Twitter"
email_address: "press@twitter.com"

```

## 02. Obtén la fuente de cada tweet.

```

{
  project: {
    source: 1
  }
}
PROJECT {source:1}
SORT
COLLATION
SKIP 0
LIMIT 0
VIEW [ ] [ ] [ ]
Displaying documents 1 - 20 of 24

_id: ObjectId("5c8eccb0caa187d17ca623f5")
source: "web"

_id: ObjectId("5c8eccb0caa187d17ca623f7")
source: "<a href='http://www.tweetdeck.com' rel='nofollow'>TweetDeck</a>"

_id: ObjectId("5c8eccb0caa187d17ca623fa")
source: "<a href='http://blackberry.com/twitter' rel='nofollow'>Twitter for Bla..."

_id: ObjectId("5c8eccb0caa187d17ca623fc")
source: "<a href='http://www.echofon.com/' rel='nofollow'>Echofon</a>"

_id: ObjectId("5c8eccb0caa187d17ca623fe")
source: "<a href='http://83degrees.com/to/powertwitter' rel='nofollow'>Power Tw..."

```

## 03. Obtén el nombre de todas las compañías fundadas en octubre.

```

{
  filter: {
    founded_month: 10
  },
  project: {
    name: 1
  }
}

```

FILTER {founded\_month:10} OPTIONS FIND RESET ...

PROJECT {name:1}

SORT MAXTIMEMS 5000

COLLATION SKIP 0 LIMIT 0

VIEW { } Refresh

Displaying documents 1 - 20 of 301

```

_id: ObjectId("52cdef7c4bab8bd675297d8a")
name: "Wetpaint"

_id: ObjectId("52cdef7c4bab8bd675297da4")
name: "Powerset"

_id: ObjectId("52cdef7c4bab8bd675297dff")
name: "TheFind"

_id: ObjectId("52cdef7c4bab8bd675297e1e")
name: "TechnologyGuide"

_id: ObjectId("52cdef7c4bab8bd675297e22")
name: "Techmeme"

```

#### 04. Obtén el nombre de todas las compañías fundadas en 2008.

```

{
  filter: {
    founded_year: 2008
  },
  project: {
    name: 1
  }
}

```

FILTER {founded\_year:2008} OPTIONS FIND RESET ...

PROJECT {name:1}

SORT MAXTIMEMS 5000

COLLATION SKIP 0 LIMIT 0

VIEW { } Refresh

Displaying documents 1 - 20 of 1224

```

_id: ObjectId("52cdef7c4bab8bd675297da8")
name: "OpenX"

_id: ObjectId("52cdef7c4bab8bd675298218")
name: "WonderHowTo"

_id: ObjectId("52cdef7c4bab8bd675298232")
name: "First30Days"

_id: ObjectId("52cdef7c4bab8bd675298244")
name: "Mibura"

_id: ObjectId("52cdef7c4bab8bd675298259")
name: "TVosz"

```

#### 05. Obtén todos los post del autor machine.

```

{
  filter: {
    author: 'machine'
  }
}

```

**FILTER** `{author:"machine"}` **OPTIONS** **FIND** **RESET** **...**

**PROJECT**

**SORT** **MAXTIMEMS** 5000

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

**ADD DATA** **VIEW** **REFRESH** Displaying documents 1 - 20 of 500

```

_id: ObjectId("50ab0f8bbcf1bfe2536dc3f9")
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

_id: ObjectId("50ab0f8bbcf1bfe2536dc3fa")
body: "We the People of the United States, in Order to form a more perfect Un..."
permalink: "jNsgObovWyKEoXNdytis"
author: "machine"
title: "US Constitution"
> tags: Array
> comments: Array
date: 2012-11-20T05:05:15.232+00:00

_id: ObjectId("50ab0f8bbcf1bfe2536dc3fb")
body: "Four score and seven years ago our Fathers brought forth on this conti..."

```

## 06. Obtén todos los tweets provenientes de la web.

```
{
  filter: {
    source: 'web'
  }
}
```

**FILTER** `{source:"web"}` **OPTIONS** **FIND** **RESET** **...**

**PROJECT**

**SORT** **MAXTIMEMS** 5000

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

**ADD DATA** **VIEW** **REFRESH** Displaying documents 1 - 20 of 11141

```

_id: ObjectId("5c8eccb0caa187d17ca623f5")
text: "eu preciso de terminar de fazer a minha tabela, está muito foda **"
in_reply_to_status_id: null
retweet_count: null
contributors: null
created_at: "Thu Sep 02 18:11:23 +0000 2010"
geo: null
source: "web"
coordinates: null
in_reply_to_screen_name: null
truncated: false
> entities: Object
retweeted: false
place: null
> user: Object
favorited: false
in_reply_to_user_id: null
id: 22819396900

_id: ObjectId("5c8eccb0caa187d17ca623ff")
text: "First week of school is over :P"
in_reply_to_status_id: null
retweet_count: null

```

## 07. Obtén todas las compañías fundadas en octubre del 2008.

```
{
  filter: {
    $and: [
      {
        founded_month: 10
      },
      {
        founded_year: 2008
      }
    ]
  }
}
```

```

}
}
}
}

```

**FILTER** `{ $and: [{ founded_month: 10 }, { founded_year: 2008 } ] }` **OPTIONS** **FIND** **RESET** **\*\***

**PROJECT**

**SORT** **MAXTIME** 5000

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

**ADD DATA** **VIEW** **{} {} {}** **Displaying documents 1 - 20 of 63** **REFRESH**

```

_id: ObjectId("52cdef7c4bab8bd6752985ca")
name: "tunesBag"
permalink: "tunesbag"
crunchbase_url: "http://www.crunchbase.com/company/tunesbag"
homepage_url: "http://www.tunesBag.com"
blog_url: "http://tunesBag.blogspot.com/"
blog_feed_url: "http://blog.tunesbag.com/feeds/posts/default?alt=rss"
twitter_username: ""
category_code: "games_video"
number_of_employees: null
founded_year: 2008
founded_month: 10
founded_day: 1
deadpooled_year: null
deadpooled_month: null
deadpooled_day: null
deadpooled_url: null
tag_list: "music, cloud, locker, mp3, music-streaming, streaming"
alias_list: ""
email_address: "office@tunesBag.com"
phone_number: "+43 680 215 27 96"
description: "Social Music Player"

```

## 08. Obtén todas las compañías con más de 50 empleados.

```

{
  filter: {
    number_of_employees: {
      $gt: 50
    }
  },
  project: {
    name: 1,
    number_of_employees: 1
  }
}

```

FILTER

{number\_of\_employees:{\$gt:50}}

OPTIONS

PROJECT

{name:1,number\_of\_employees:1}

SORT

MAXTIME

5000

COLLATION

SKIP

0

LIMIT

0

VIEW

Displaying documents 1 - 20 of 793

\_id: ObjectId("52cdef7c4bab8bd675297d8e")

name: "Facebook"

number\_of\_employees: 5299

\_id: ObjectId("52cdef7c4bab8bd675297d94")

name: "Twitter"

number\_of\_employees: 1300

\_id: ObjectId("52cdef7c4bab8bd675297da2")

name: "Cisco"

number\_of\_employees: 63000

\_id: ObjectId("52cdef7c4bab8bd675297da3")

name: "Yahoo!"

number\_of\_employees: 13600

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

```

{
  filter: {
    $and: [
      {
        comments: {
          $gte: 10
        }
      },
      {
        comments: {
          $lte: 30
        }
      }
    ]
  },
  project: {
    title: 1,
    comments: 1
  }
}

```



FILTER

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

OPTIONS

PROJECT

{ title: 1, comments: 1 }

SORT

MAXTIMEMS 5000

COLLATION

SKIP 0

LIMIT 0

VIEW

{ }

Displaying documents 1 - 20 of 1930

\_id: ObjectId("4ba267dc238d3ba3ca000006")

title: "11 Amazing Treehouses from Around the World"

comments: 15

\_id: ObjectId("4ba267dc238d3ba3ca00000b")

title: "NASA - The Wizard Nebula "

comments: 14

\_id: ObjectId("4ba267dc238d3ba3ca00000c")

title: "WISE Captures a Cosmic Rose"

comments: 12

\_id: ObjectId("4ba267dc238d3ba3ca000012")

title: "Chris Parnell, Horatio Sanz Cast in Ferrell/Heder Sitcom "

comments: 22

**010.** *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
}

```

**FILTER** `{($and:[{number_of_employees:{$ne:null}}, {number_of_employees:{$ne:0}}])}` **OPTIONS** **FIN**

**PROJECT** `{name:1,number_of_employees:1}`

**SORT** `{number_of_employees:1}` **MAXTIMEMS** 5000

**COLLATION** **SKIP** 0 **LIMIT** 1

VIEW `{}` `[]` `[ ]` `[ ]` Displaying documents 1 - 1 of 1

```

_id: ObjectId("52cdef7c4bab8bd675297e60")
name: "FeVote"
number_of_employees: 1

```

**011.** *Obtén la empresa con el mayor número de empleados.*

```

{
  project: {
    name: 1,
    number_of_employees: 1
  },
  sort: {
    number_of_employees: -1
  },
  limit: 1
}

```

**FILTER** **OPTIONS**

**PROJECT** `{name:1,number_of_employees:1}`

**SORT** `{number_of_employees:-1}` **MAXTIMEMS** 5000

**COLLATION** **SKIP** 0 **LIMIT** 1

VIEW `{}` `[]` `[ ]` `[ ]` Displaying documents 1 - 1 of 1

```

_id: ObjectId("52cdef7c4bab8bd67529856a")
name: "IBM"
number_of_employees: 388000

```

**012.** *Obtén la historia más comentada.*

```

{
  project: {
    title: 1,
    comments: 1
  },
  sort: {
    comments: -1
  },
  limit: 1
}

```

**FILTER** **OPTIONS**

**PROJECT** `{title:1,comments:1}`

**SORT** `{comments:-1}` **MAXTIMEMS** 5000

**COLLATION** **SKIP** 0 **LIMIT** 1

VIEW `{}` `[]` `[ ]` `[ ]` Displaying documents 1 - 1 of 1

```

_id: ObjectId("4ba27ea0238d3ba3ca002251")
title: "Republican Brown wins Massachusetts Senate seat!"
comments: 1864

```

**013.** *Obtén la historia menos comentada.*

```

{

```

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

PROJECT

{title:1,comments:1}

SORT

{comments:1}

MAXTIME

5000

COLLATION

SKIP

0

LIMIT

1

VIEW

Displaying documents 1 - 1 of 1

\_id: ObjectId("4ba27e1a238d3ba3ca002161")

title: "UA Tech Park chosen for \$32 million 'Solar Zone' project"

comments: 0

# SESIÓN 05

## RETO 1

### 01. Propiedades que no permitan fiestas.

```
{
  filter: {
    house_rules: RegExp('no parties', i)
  }
}
```

Displaying documents 1 - 20 of 243

```
{
  "_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
}
```

SHOW 14 MORE FIELDS

### 02. Propiedades que admitan mascotas.

```
{
  filter: {
    house_rules: RegExp('pets allowed', i)
  }
}
```

Displaying documents 1 - 20 of 59

```
{
  "_id": "11197217",
  "listing_url": "https://www.airbnb.com/rooms/11197217",
  "name": "2 Bedroom Ipanema-Copacabana",
  "summary": "Our apartment is strategically located between the beaches of Copacaba...",
  "space": "",
  "description": "Our apartment is strategically located between the beaches of Copacaba...",
  "neighborhood_overview": "",
  "notes": "",
  "transit": "",
  "access": "",
  "interaction": "",
  "house_rules": "We would appreciate if the guest, as leaving the apartment even for a ...",
  "property_type": "Apartment",
  "room_type": "Entire home/apt",
  "bed_type": "Real Bed",
  "minimum_nights": "4",
  "maximum_nights": "30",
  "cancellation_policy": "moderate",
  "last_scraped": "2019-02-11T05:00:00.000+00:00",
  "calendar_last_scraped": "2019-02-11T05:00:00.000+00:00",
  "first_review": "2016-08-23T04:00:00.000+00:00",
  "last_review": "2018-02-15T05:00:00.000+00:00",
  "accommodates": 5,
  "bedrooms": 2,
  "beds": 5
}
```

SHOW 14 MORE FIELDS

### 03. Propiedades que no permitan fumadores.

```
{
  filter: {
    house_rules: RegExp('no smoking', i)
  }
}
```

**FILTER** {house\_rules: /no smoking/i} **OPTIONS**

**ADD DATA** **VIEW** **{}** **TABLE**

Displaying documents 1 - 20 of 639

```

_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**

#### 04. Propiedades que no permitan fiestas ni fumadores.

```

{
  filter: {
    house_rules: RegExp('no smoking|no parties', i)
  }
}

```

ADD

VIEW

Displaying documents 1 - 20 of 758

DATA

{}

```

_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_overvi_ : "Great neighborhood - many terrific restaurants, bakeries, bagelr...
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

```

_id: "10083468"
listing_url: "https://www.airbnb.com/rooms/10083468"
name: "Be Happy in Porto"
summary: "Be Happy Apartment is an amazing space. Renovated and comfortable apar..."
space: "Be Happy Apartment is housed in a typical Porto building, where the ap..."
description: "Be Happy Apartment is an amazing space. Renovated and comfortable apar..."
neighborhood_overvi_ : ""Be happy in Porto" offers a quality accommodation to spend a fi
notes: ""
transit: "When you arrive at Airport Francisco Sá Carneiro my advice to take the "

```

## RETO 2

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'
  }
}

```

**FILTER** [ number\_of\_reviews: {\$gte: 50}, "review\_scores.review\_scores\_rating": {\$gte: 80}, amenities: {\$in: [/Ethernet/i]}, "ad" ] **OPTIONS** **FILE**

**ADD DATA** **VIEW** [ ] [ ] [ ]

Displaying documents 1 - 6 of 6

```

{
  "_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 bicycle, 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
}

```

## RETO 3

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
}
}
}]

```

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

```

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

```

```

{
  "_id": "10006546",
  "listing_url": "https://www.airbnb.com/rooms/10006546",
  "name": "Ribeira Charming Duplex",
  "summary": "Fantastic duplex apartment with three bedroom located in the histori...",
  "space": "Privileged views of the Douro River and Ribeira square, our apartment ...",
  "description": "Fantastic duplex apartment with three bed located in the histori..."
}

```

```

{
  "_id": "10009999",
  "listing_url": "https://www.airbnb.com/rooms/10009999",
  "name": "Horto flat with small garden",
  "summary": "One bedroom + sofa-bed in quiet a neighbourhood right next t...",
  "space": "Lovely one bedroom + sofa-bed in th perfect for two but ...",
  "description": "One bedroom + sofa-bed in qui neighbourhood right next t..."
}

```

**\$group** **Output after \$group stage** (Sample of 1 document)

```

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 }

```

```

{
  "_id": null,
  total: 5303
}

```

## PROYECTO

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: [/Ethernet/i]}
}}, {$match: {

```





## SESIÓN 06

### 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: {
  costo_promedio: {
    $divide: [
      '$total',
      '$recamaras'
    ]
  }
}}, {$project: {
  costo_promedio: 1
}}]
```

\$project

Output after [\\$project](#) stage ⓘ (Sample of 9 documents)

```
1 {
2   costo_promedio: 1
3 }
```

```
_id: "Turkey"
costo_promedio: 209.875
```

### RETO 2

Usando las colecciones *comments* y *users*, se requiere conocer el correo y contraseña de cada persona que realizó un comentario. Construye un pipeline que genere como resultado estos datos.

**NO CIERES ESTE PIPELINE PUES LO USAREMOS MÁS ADELANTE**

```
[[{$lookup: {
  from: 'users',
  localField: 'name',
  foreignField: 'name',
  as: 'usuario'
}}, {$addFields: {
  usuario_objeto: {$arrayElemAt: ['$usuario', 0]}
}}, {$addFields: {
  usuario_password: '$usuario_objeto.password'
}}, {$project: {
```

\_id:0,  
name:1,  
email:1,  
usuario\_password:1  
}}}

\$project

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

```
1 /*
2  * specifications: The fields to
3  *   include or exclude.
4  */
5 {
6   _id:0,
7   name:1,
8   email:1,
9   usuario_password:1
10 }
```

name: "Andrea Le"  
email: "andrea\_le@fakegmail.com"  
usuario\_password: "\$2b\$12\$3587HwL2y0P1E6kYrcbKOKx22.wsKEdLt50F734/vKdhudLM8Ve"

name: "Greg Powell"  
email: "greg\_powell@fakegmail.com"  
usuario\_password: "\$2b\$12\$XpveUB6kIiU3zG5aABw260itIB7cD8bSUNJAz4wDF4XXyNNJ/mp76"

name: "Talisa Maegyr"  
email: "oona\_chaplin@gameofthron.es"  
usuario\_password: "\$2b\$12\$83DKrkTVNJD1gToXClpcw0JvnKICuSLcrTXNfZIEoqbHIQURsNrXS"

RETO 3

Usando el pipeline que generaste en el Reto 2, genera la vista correspondiente.  
sample\_mflix.4YCM (view on: sample\_mflix.comments) [MODIFY SOURCE](#)

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

FILTER

VIEW

Displaying documents

name: "Andrea Le"  
email: "andrea\_le@fakegmail.com"  
password: "\$2b\$12\$3587HwL2y0P1E6kYrcbKOKx22.wsKEdLt50F734/vKdhudLM8Ve"

name: "Greg Powell"  
email: "greg\_powell@fakegmail.com"  
password: "\$2b\$12\$XpveUB6kIiU3zG5aABw260itIB7cD8bSUNJAz4wDF4XXyNNJ/mp76"

name: "Talisa Maegyr"  
email: "oona\_chaplin@gameofthron.es"  
password: "\$2b\$12\$83DKrkTVNJD1gToXClpcw0JvnKICuSLcrTXNfZIEoqbHIQURsNrXS"

name: "Cameron Duran"  
email: "cameron\_duran@fakegmail.com"  
password: "\$2b\$12\$50w2j63ATGmhVoh2rgdjv.w0d9TV0Jb9Xk/Anms0fxVSvGMf5MvvK"

name: "Petyr Baelish"  
email: "aidan\_gillen@gameofthron.es"  
password: "\$2b\$12\$qM.YvmiekyYY7p7phpK30icbRCDKw7ESwYAnG/o9YnfHC0Mhkmbi"

sample\_airbnb.4YCM\_RETO03 (view on: sample\_airbnb.listingsAndReviews) [MODIFY SOURCE](#)

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

FILTER

OPTIONS

VIEW

Displaying documents 1 - 9 of 9

\_id: "Australia"  
costo: 107.8

\_id: "Brazil"  
costo: 348.7

\_id: "Canada"  
costo: 115.0

\_id: "China"  
costo: 393.0

\_id: "Hong Kong"  
costo: 514.6

\_id: "Portugal"  
costo: 40.0

PROYECTO

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:

```
[$project: {
  _id:0,
  genres:1,
  countries:1,
}], {$unwind: {
  path: "$genres",
  preserveNullAndEmptyArrays:false
}], {$unwind: {
  path: "$countries",
  preserveNullAndEmptyArrays:false
}], {$group: {
  _id: "$countries",
  total: {$push:{
    genero:"$genres"
  }}
}], {$unwind: {
  path: "$total",
  preserveNullAndEmptyArrays:false
}], {$addFields: {
  genero:"$total.genero"
}], {$group: {
  _id:{pais:"$_id",
    genero:"$genero"
  },suma:{$sum:1}
}], {$addFields: {
  pais:"$_id.pais",
  genero:"$_id.genero"
}], {$project: {
  _id:0
}], {$sort: {
  pais:1
}}]
```

	suma Int32	pais String	genero String
1	1	"Algeria"	"Musical"
2	1	"Albania"	"Comedy"
3	4	"Afghanistan"	"Documentary"
4	1	"Algeria"	"Mystery"
5	2	"Algeria"	"Crime"
6	1	"Albania"	"War"
7	1	"Algeria"	"Adventure"
8	3	"Afghanistan"	"History"
9	4	"Algeria"	"War"
10	1	"Afghanistan"	"Action"
11	13	"Algeria"	"Drama"
12	3	"Algeria"	"History"
13	1	"Albania"	"Crime"
14	1	"Algeria"	"Comedy"
15	3	"Afghanistan"	"War"
16	1	"Albania"	"Thriller"
17	1	"Afghanistan"	"News"
18	4	"Afghanistan"	"Drama"
19	5	"Albania"	"Drama"
20	1	"Algeria"	"Thriller"

## SESIÓN 07

### RETO 1

- 01.** Usando como base el archivo `movies.dat`, limpiarlo e importar los datos en la tabla `movies` creada en el Reto 1.
- 02.** Usando como base el archivo `ratings.dat`, limpiarlo e importar los datos en la tabla `ratings` creada en el Reto 2.

### RETO 2

- 01.** Usando como base el archivo `movies.dat`, limpiarlo e importar los datos en la tabla `movies` creada en el Reto 1.
- 02.** Usando como base el archivo `ratings.dat`, limpiarlo e importar los datos en la tabla `ratings` creada en el Reto 2.

### RETO 3

- 01.** Usando Se deberá de crear colecciones e importar los datos de los archivos `movies.csv` y `ratings.csv`.

USERS:

Configure Import Settings

Detected file format: csv

Encoding:

Columns:

<input checked="" type="checkbox"/> Source Column	Field Type
<input checked="" type="checkbox"/> id	<input type="text" value="int"/>
<input checked="" type="checkbox"/> genero	<input type="text" value="text"/>
<input checked="" type="checkbox"/> edad	<input type="text" value="int"/>
<input checked="" type="checkbox"/> ocup	<input type="text" value="int"/>
<input checked="" type="checkbox"/> cp	<input type="text" value="text"/>


id	genero	edad	ocup	cp
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

MOVIES:

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"/> id	<span>int</span>
<input checked="" type="checkbox"/> title	<span>text</span>
<input checked="" type="checkbox"/> generos	<span>text</span>


id	title	generos
1	Toy Story (...)	Animation ...
2	Jumanji (19...	Adventure ...
3	Grumpier O...	Comedy Ro...
4	Waiting to ...	Comedy Dr...
5	Father of th...	Comedy

< Back   Next >   Cancel

## RATINGS:

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	<span>int</span>
<input checked="" type="checkbox"/> movieid	<span>int</span>
<input checked="" type="checkbox"/> rating	<span>int</span>
<input checked="" type="checkbox"/> time_stamp	<span>int</span>

userid	movieid	rating	time_stamp
1	1193	5	978300760
1	661	3	978302109
1	914	3	978301968
1	3408	4	978300275
1	2355	5	978824291

< Back   Next >   Cancel

## PROYECTO

*A continuación se realizaran algunas operaciones de agregar, modificar y eliminar un documento JSON en una Colección.*

**01.** Agregar los siguientes registros en formato CSV a la Colección *movies*

*4000, Avengers: Endgame (2019), Fantasy|Sci-Fi*

*4001, Glass (2019), Drama|Fantasy*

**FILTER** {id:{\$in:["4000","4001"]}}

**PROJECT**

**SORT**

**COLLATION**

**ADD DATA**



VIEW



```
_id: ObjectId("5f18f66e787ea03e34563e0e")
id: "4000"
title: "Avengers: Endgame (2019)"
generos: "Fantasy|Sci-Fi"
```

```
_id: ObjectId("5f18f66e787ea03e34563e0f")
id: "4001"
title: "Glass (2019)"
generos: "Drama|Fantasy"
```

## SESIÓN 08

### RETO 1

**01. Descarga la fuente de datos de los locales de Starbucks:**

[directory.csv](#)

**02. Analiza los datos, limpia los datos en caso de ser necesario.**

**03. Elige MySQL o MongoDB y crea una base de datos para el conjunto de datos del reto.**

**04. Carga los datos en la base de datos que elegiste y revisa que éstos se muestren correctamente.**

**05. Usando la latitud y longitud de tu posición actual, encuentra el Starbucks más cercano a tu posición. Para conocer tu posición actual puedes usar Google Maps para, sólo debes copiar los datos de la URL.**

```
[$match: {
  "City": "Mexico"
}], {$match: {
  $and: [
    {Longitude: {$lte: -99.15}},
    {Longitude: {$gte: -99.17}},
    {Latitude: {$lte: 19.41}},
    {Latitude: {$gt: 19.40}}
  ]
}}
```

The screenshot shows a MongoDB query interface. On the left, a query is written in a text editor: `1 /**`  
`2 * query: The query in MQL.`  
`3 */`  
`4 {`  
`5 $and:`  
`6 [{Longitude: {$lte: -99.15}},`  
`7 {Longitude: {$gte: -99.17}},`  
`8 {Latitude: {$lte: 19.41}},`  
`9 {Latitude: {$gt: 19.40}}]`  
`10 }`

On the right, the output of the query is displayed. It shows a single document with the following fields: `Street Address: "Monterrey n.330, Col., Roma Sur Distri`  
`City: "Mexico"`  
`State/Province: "DIF"`  
`Country: "MX"`  
`Postcode: "6760"`  
`Phone Number: ""`  
`Timezone: "GMT-06:00 America/Mexico_City"`  
`Longitude: -99.16`  
`Latitude: 19.41`

### RETO 2

**01. Descarga la fuente de datos sobre la pandemia del 2009 (H1N1):**

[Pandemic \(H1N1\) 2009.csv](#)

**02. Analiza los datos, limpia los datos en caso de ser necesario.**

**03. Elige MySQL o MongoDB y crea una base de datos para el conjunto de datos del reto.**

**04. Carga los datos en la base de datos que elegiste y revisa que éstos se muestren correctamente.**

**05. Responde a las siguientes preguntas usando consultas:**

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

`select Country,Deaths from h1n1 order by Deaths DESC LIMIT 1;`

Country	Deaths
Argentina	60

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

`SELECT Country,Deaths FROM h1n1 ORDER BY Deaths ASC LIMIT 1;`

Country	Deaths
Algeria	0

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

`SELECT Country,Cases FROM h1n1 ORDER BY Cases DESC LIMIT 1;`

Country	Cases
Canada	7983

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

`SELECT Country,Cases FROM h1n1 ORDER BY Cases LIMIT 1;`



	Country	Cases
▶	Bermuda, UKOT	1

- e. **¿Cuál fue el número de muertes promedio?**

SELECT AVG(Deaths) AS PromMuertes FROM h1n1;

	PromMuertes
▶	2.9250

- f. **¿Cuál fue el número de casos promedio?**

SELECT AVG(Cases) AS PromMuertes FROM h1n1;

	PromMuertes
▶	700.7750

- g. **Top 5 de países con más muertes**

select Country,Deaths from h1n1 order by Deaths DESC LIMIT 5;

	Country	Deaths
▶	Argentina	60
	Canada	25
	Chile	14
	Australia	10
	Costa Rica	3

- h. **Top 5 de países con menos muertes**

select Country,Deaths from h1n1 order by Deaths LIMIT 5;

	Country	Deaths
▶	Algeria	0
	Antigua and Barbuda	0
	El Salvador	0
	Estonia	0
	Austria	0

### RETO 3

**01. Descarga la fuente de datos de los casos sobre la pandemia del COVID-19:**

[2019-nCoV-cases-JHU.csv](#)

**02. Analiza los datos, limpia los datos en caso de ser necesario.**

**03. Elige MySQL o MongoDB y crea una base de datos para el conjunto de datos del reto.**

**04. Carga los datos en la base de datos que elegiste y revisa que éstos se muestren correctamente.**

**05. Responde a las siguientes preguntas usando consultas:**

- a. **¿Cuál es país con mayor número de casos?**

```
[{$group: {
  _id: "$Region",
  maxCasesPerCountry: {$max: "$Confirmed"}}, {$sort: {
  maxCasesPerCountry: -1
}}, {$limit: 1
}]
```

▼

\$limit

🔴

🗑️

+

Output after [\\$limit](#) stage ⓘ (Sample)

```
1 ▾ /**
2   * Provide the number of documents to Limit.
3   */
4   1
5   |
```

```
_id: "Mainland China"
maxCasesPerCountry: 67217
```

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

```
[{$group: {
  _id: "$Region",
  maxDeathsPerCountry: {$max: "$Deaths"}}, {$sort: {
  maxDeathsPerCountry: -1
}}, {$limit: 1
}]
```

\$limit

Output after `$limit` stage ⓘ (Sample of 1 docum

```
/**
 * Provide the number of documents to limit.
 */
1
```

```
_id: "Mainland China"
maxDeathsPerCountry: 2835
```

c. **Usando las coordenadas, encuentra el epicentro del virus.**

```
[{$match: {
  Lat: { $ne: "" },
  Long: { $ne: "" }
}}, {$addFields: {
  Lat: { $convert: { input: "$Lat", to: "double"}},
  Long: { $convert: { input: "$Long", to: "double"}}
}}, {$group: {
  _id: null,
  size: {
    $sum: 1
  },
  sumLat: {
    $sum: '$Lat'
  },
  sumLong: {
    $sum: '$Long'
  },
  avgLat: {
    $avg: '$Lat'
  },
  avgLong: {
    $avg: '$Long'
  }
}}, {$project: {
  checkAvgLat: {
    $divide: [
      '$sumLat', '$size'
    ]
  },
  checkAvgLong: {
    $divide: [
      '$sumLong', '$size'
    ]
  }
}}]
```

\$project

Output after \$project stage ⓘ (Sample of 1 dc)

```

1  {
2    checkAvgLat: {
3      $divide: [
4        '$sumLat',
5        '$size'
6      ]
7    },
8    checkAvgLong: {
9      $divide: [
10       '$sumLong',
11       '$size'
12     ]
13   }
14 }

```

```

_id: null
checkAvgLat: 2.808041898492249
checkAvgLong: 2.631917158632406

```

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

```

[{$match: {
  Lat: {
    $ne: ""
  },
  Long: {
    $ne: ""
  }
}}, {$group: {
  _id: "$Region", Lat:
  {
    $max: "$Lat"
  },
  Long: {
    $max: "$Long"
  }
}}, {$addFields: {
  Lat: {
    $convert: { input:
    '$Lat',
    to: 'double'
  }
  },
  Long: {
    $convert: { input:
    '$Long', to: 'double'
  }
  }
}}, {$match: {
  $and: [
    {
      Lat: {
        $gte: 21
      }
    },
    {
      Lat: {
        $lte: 35
      }
    },
    {
      Long: {
        $gte: 30
      }
    },
    {
      Long: {
        $lte: 45
      }
    }
  ]
}}]

```

```
}
}
]
}}, {$limit: 5}, {$project: {
_id: 1
}}]
```

||  \$match ☒

Output after [\\$match](#) stage ⓘ (Sample of 6 documents)

```
1 {
2   $and: [
3     {
4       Lat: {
5         $gte: 21
6       }
7     },
8     {
9       Lat: {
10        $lte: 35
11      }
12    }
13   ]
14 }
```

\_id: "Egypt"  
Lat: 26  
Long: 30

\_id: "Lebanon"  
Lat: 33.8547  
Long: 35.8623

\_id: "Saudi Arabia"  
Lat: 24  
Long: 45