

LABORATORIO 16

CONSULTAS SQL CON FUNCIONES AGREGADAS

Adriana Paola Salinas García A01703675

1. Consulta de una tabla completa:

`SELECT * FROM Materiales`

	Clave	Descripcion	Costo
1	1000	Vanilla 3/16	100.00
2	1010	Vanilla 4/32	115.00
3	1020	Vanilla 3/17	130.00

Número de Tuplas: 44

2. Selección:

`SELECT * FROM Materiales`
`Where clave = 1000`

	Clave	Descripcion	Costo
1	1000	Vanilla 3/16	100.00

Número de Tuplas: 1

3. Proyección:

`SELECT Clave, RFC, Fecha FROM Entregan`

	Clave	RFC	Fecha
1	1000	AAAA800101	1998-07-08 00:00:00.000
2	1000	AAAA800101	1999-08-08 00:00:00.000
3	1000	AAAA800101	2000-04-06 00:00:00.000

Número de Tuplas: 132

4. Reunión Natural:

`SELECT * FROM Materiales, Entregan`
`WHERE Materiales.Clave = Entregan.Clave`

	Clave	Descripcion	Costo	Clave	RFC	Numero	Fecha	Cantidad
1	1000	Vanilla 3/16	100.00	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00
2	1000	Vanilla 3/16	100.00	1000	AAAA800101	5019	1999-08-08 00:00:00.000	254.00
3	1000	Vanilla 3/16	100.00	1000	AAAA800101	5019	2000-04-06 00:00:00.000	7.00

Número de Tuplas: 132

Si algún material no ha se ha entregado ¿Aparecería en el resultado de esta consulta? **No**

5. Reunión con criterio específico:

```
SELECT * FROM Entregan, Proyectos
WHERE Entregan.Numero <= Proyectos.Numero
```

	Clave	RFC	Numero	Fecha	Cantidad	Numero	Denominacion
1	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00	5000	Vamos Mexico
2	1200	EEEE800101	5000	2000-03-05 00:00:00.000	177.00	5000	Vamos Mexico
3	1400	AAAA800101	5000	2002-03-12 00:00:00.000	382.00	5000	Vamos Mexico

Número de Tuplas: 1188

6. Unión (junto con selección):

```
(SELECT * FROM Entregan where Clave = 1450)
union
(SELECT * FROM Entregan where Clave = 1300)
```

	Clave	RFC	Numero	Fecha	Cantidad
1	1300	G G G G 800101	5005	2002-06-10 00:00:00.000	521.00
2	1300	G G G G 800101	5005	2003-02-02 00:00:00.000	457.00
3	1300	G G G G 800101	5010	2003-01-08 00:00:00.000	119.00

Número de Tuplas: 3

¿Cuál sería una consulta que obtuviera el mismo resultado sin usar el operador Unión?

```
SELECT * FROM Entregan
WHERE Clave = 1450 OR Clave = 1300
```

7. Intersección (junto con selección y proyección):

```
(SELECT Clave FROM Entregan
WHERE Numero = 5001)
INTERSECT
(SELECT Clave FROM Entregan
WHERE Numero = 5018)
```

Con Subconsulta:

```
SELECT Clave FROM Entregan
WHERE Numero = 5001 and Clave
IN
(SELECT Clave FROM Entregan
WHERE Numero = 5018 )
```

	Clave
1	1010

Número de Tuplas: 1

8. Diferencia (junto con selección):

```
SELECT * FROM Entregan
WHERE Clave NOT IN
(SELECT Clave FROM Entregan
WHERE Clave = 1000)
```

	Clave	RFC	Numero	Fecha	Cantidad
1	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00
2	1010	BBBB800101	5018	2000-11-10 00:00:00.000	667.00
3	1010	BBBB800101	5018	2002-03-29 00:00:00.000	523.00

Número de Tuplas: 129

9. Producto Cartesiano:

```
SELECT * FROM Entregan, Materiales
```

	Clave	RFC	Numero	Fecha	Cantidad	Clave	Descripcion	Costo
1	1000	AAA800101	5000	1998-07-08 00:00:00.000	165.00	1000	Vanilla 3/16	100.00
2	1000	AAA800101	5019	1999-08-08 00:00:00.000	254.00	1000	Vanilla 3/16	100.00
3	1000	AAA800101	5019	2000-04-06 00:00:00.000	7.00	1000	Vanilla 3/16	100.00

Número de Tuplas: 5808

10. A partir de una especificación:

```
set dateformat dmy
SELECT Descripcion FROM Materiales, Entregan
WHERE Materiales.Clave = Entregan.Clave AND
Fecha BETWEEN '01/01/00' AND '31/12/00'
```

	Descripcion
1	Vanilla 3/16
2	Vanilla 4/32
3	Vanilla 4/32

Número de Tuplas: 28

¿Por qué aparecen varias veces algunas descripciones de material?
Porque se entregó el material en diferentes fechas.

11. Calificador distinct:

```
SELECT DISTINCT Descripcion FROM Materiales, Entregan
WHERE Materiales.Clave = Entregan.Clave AND
Fecha BETWEEN '01/01/00' AND '31/12/00'
```

	Descripcion
1	Arena
2	Block
3	Cantera rosa

Número de Tuplas: 22

¿Qué resultado obtienes en esta ocasión?
No se repite la descripción de los materiales. No se repiten tuplas.

12. Ordenamientos:

```
SELECT Proyectos.Numero, Denominacion, Fecha, Cantidad
FROM Entregan, Proyectos
WHERE Entregan.Numero = Proyectos.Numero
ORDER BY Proyectos.Numero ASC, Fecha DESC
```

	Numero	Denominacion	Fecha	Cantidad
1	5000	Vamos Mexico	2002-03-12 00:00:00.000	382.00
2	5000	Vamos Mexico	2000-03-05 00:00:00.000	177.00
3	5000	Vamos Mexico	1998-07-08 00:00:00.000	165.00

Número de Tuplas: 132

13. Operadores de cadena:

a)

```
SELECT * FROM Materiales
WHERE Descripcion LIKE 'Si%'
```

	Clave	Descripcion	Costo
1	1120	Sillar rosa	100.00
2	1130	Sillar gris	110.00

Número de Tuplas: 2

¿Qué resultado obtienes?

Un listado de materiales que empiezan con “si”

Explica que hace el símbolo '%’.

Buscar elementos que tengan coincidencia con los caracteres dados.

¿Qué sucede si la consulta fuera: LIKE 'Si' ?

Busca una palabra “Si”

¿Qué resultado obtienes?

Ninguno

Explica a qué se debe este comportamiento.

Solo buscaría elementos llamados “Si”.

b)

```
DECLARE @foo varchar(40);
DECLARE @bar varchar(40);
SET @foo = '¿Que resultado?';
SET @bar = '¿¿¿???'
SET @foo += ' obtienes?';
PRINT @foo + @bar;
```

¿Qué resultado obtienes de ejecutar el código?
Se obtiene el mensaje: ¿Que resultado obtienes? ¿¿¿???

¿Para qué sirve DECLARE?
Define el tipo de dato de las variables

¿Cuál es la función de @foo?
La arroba designa que es una variable, declara la variable foo como un string que luego se concatena con otro string.

¿Que realiza el operador SET?
Asigna un valor a las variables.

- c) Ahora explica el comportamiento, función y resultado de cada una de las siguientes consultas:

`SELECT RFC FROM Entregan WHERE RFC LIKE '[A-D]%'`;

Hace una selección del RFC que empiece ya sea con la letra A, B, C o D.

	RFC
1	AAAA800101
2	AAAA800101
3	AAAA800101

Número de Tuplas: 72

`SELECT RFC FROM Entregan WHERE RFC LIKE '[^A]%'`;

Hace una selección del RFC que no empiece con la letra A.

	RFC
1	BBBB800101
2	BBBB800101
3	BBBB800101

Número de Tuplas: 114

`SELECT Numero FROM Entregan WHERE Numero LIKE '___6'`;

Busca números que terminen con 6, los _ son comodines, en este caso son tres por lo que habrá 3 números antes que el 6.

	Numero
1	5016
2	5016
3	5006

Número de Tuplas: 14

14. Operadores lógicos:

a) Between

```
SELECT Clave, RFC, Numero, Fecha, Cantidad
FROM Entregan
WHERE Numero Between 5000 and 5010;
```

	Clave	RFC	Numero	Fecha	Cantidad
1	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00
2	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00
3	1020	CCCC800101	5002	2001-07-29 00:00:00.000	582.00

Número de Tuplas: 60

¿Cómo filtrarías rangos de fechas?

```
SELECT Clave, RFC, Numero, Fecha, Cantidad
FROM Entregan
WHERE Fecha Between '01-JAN-2000' and '31-DEC-2000';
```

b) Exists

```
SELECT RFC,Cantidad, Fecha,Numero
FROM [Entregan]
WHERE [Numero] Between 5000 and 5010 AND
Exists (SELECT [RFC]
FROM [Proveedores]
WHERE RazonSocial LIKE 'La%' and [Entregan].[RFC] = [Proveedores].[RFC])
```

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165.00	1998-07-08 00:00:00.000	5000
2	CCCC800101	582.00	2001-07-29 00:00:00.000	5002
3	AAAA800101	86.00	1999-01-12 00:00:00.000	5008

Número de Tuplas: 16

¿Qué hace la consulta?

Selecciona el RFC, Cantidad, fecha, numero de las entregas que tienen número entre 5010 y 5000 y el RFC en la razón social empiece en “La”

¿Qué función tiene el paréntesis () después de EXISTS?

Que la consulta también tenga el RFC en donde la razón social empiece con “La”

15. Operador IN:

```
SELECT RFC,Cantidad, Fecha,Numero
FROM [Entregan] e
WHERE [Numero] Between 5000 and 5010 AND e.RFC
IN (SELECT [RFC]
FROM [Proveedores]
WHERE RazonSocial LIKE 'La%')
```

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165.00	1998-07-08 00:00:00.000	5000
2	CCCC800101	582.00	2001-07-29 00:00:00.000	5002
3	AAAA800101	86.00	1999-01-12 00:00:00.000	5008

Número de Tuplas: 16

16. Tomando de base la consulta anterior del EXISTS, realiza el query que devuelva el mismo resultado, pero usando el operador NOT IN Realiza un ejemplo donde apliques algún operador: ALL, SOME o ANY.

```
SELECT RFC,Cantidad, Fecha,Numero
FROM [Entregan] e
WHERE [Numero] Between 5000 and 5010 AND e.RFC
NOT IN (SELECT [RFC]
FROM [Proveedores]
WHERE RazonSocial NOT LIKE 'La%')
```

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165.00	1998-07-08 00:00:00.000	5000
2	CCCC800101	582.00	2001-07-29 00:00:00.000	5002
3	AAAA800101	86.00	1999-01-12 00:00:00.000	5008

Número de Tuplas: 16

17. ¿Qué hace la siguiente sentencia? Explica por qué.

```
SELECT TOP 2 * FROM Proyectos
```

	Numero	Denominacion
1	5000	Vamos Mexico
2	5001	Aztecón

Número de Tuplas: 2

Muestra las primeras dos tuplas de la tabla Proyectos

18. ¿Qué sucede con la siguiente consulta? Explica por qué.

```
SELECT TOP Numero FROM Proyectos
```

Msg 102, Level 15, State 1, Line 137
Incorrect syntax near 'Numero'.

Muestra un error porque no se especifica cuantas filas se quieren extraer.

19. Modificar la estructura de una tabla existente:

```
ALTER TABLE materiales ADD PorcentajeImpuesto NUMERIC(6,2);  
UPDATE materiales SET PorcentajeImpuesto = 2*clave/1000;
```

(44 rows affected)

¿Qué consulta usarías para obtener el importe de las entregas, es decir, el total en dinero de lo entregado, basado en la cantidad de la entrega y el precio del material y el impuesto asignado?

```
SELECT E.Clave, E.Cantidad*(M.Costo*(PorcentajeImpuesto/100)+1) as "Total"  
FROM Materiales M, Entregan E  
WHERE M.Clave=E.Clave
```

	Clave	Total
1	1000	495.0000000000
2	1000	762.0000000000
3	1000	21.0000000000
4	1000	475.5000000000

Número de Tuplas: 132

20. Creación de vistas:

a)

```
CREATE VIEW Menor AS (  
SELECT * FROM Materiales  
WHERE Clave = 1000  
)
```

```
SELECT * FROM Menor
```

	Clave	Descripcion	Costo	PorcentajeImpuesto
1	1000	Vanilla 3/16	100.00	2.00

Número de Tuplas: 1

b)

```
CREATE VIEW EmpiezanAD AS (  
SELECT RFC FROM Entregan  
WHERE RFC LIKE '[A-D]%'  
)
```

Results		Messages	
	RFC		
1	AAAA800101		
2	AAAA800101		
3	AAAA800101		

Número de Tuplas: 72

c)

```
CREATE VIEW MaterialSI AS (  
SELECT * FROM Materiales  
WHERE Descripcion LIKE 'Si%'  
)  
SELECT * FROM MaterialSI
```

	Clave	Descripcion	Costo	PorcentajeImpuesto
1	1120	Sillar rosa	100.00	2.24
2	1130	Sillar gris	110.00	2.26

Número de Tuplas: 2

d)

```
CREATE VIEW Numero AS(  
SELECT Clave, RFC, Numero, Fecha, Cantidad  
FROM Entregan  
WHERE Numero Between 5000 and 5010  
)  
SELECT * FROM Numero
```

	Clave	RFC	Numero	Fecha	Cantidad
1	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00
2	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00
3	1020	CCCC800101	5002	2001-07-29 00:00:00.000	582.00

Número de Tuplas: 60

e)

```
CREATE VIEW RazonLa AS(  
SELECT RFC,Cantidad, Fecha,Numero  
FROM [Entregan] e  
WHERE [Numero] Between 5000 and 5010 AND e.RFC  
IN (SELECT [RFC]  
FROM [Proveedores]  
WHERE RazonSocial LIKE 'La%')  
)  
SELECT * FROM RazonLa
```

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165.00	1998-07-08 00:00:00.000	5000
2	CCCC800101	582.00	2001-07-29 00:00:00.000	5002
3	AAAA800101	86.00	1999-01-12 00:00:00.000	5008

Número de Tuplas: 16

21. Consultas:

- 1) Los materiales (clave y descripción) entregados al proyecto "México sin ti no estamos completos".

```
SELECT E.Clave, M.Descripcion
FROM Materiales as M, Entregan as E, Proyectos as P
WHERE M.Clave=E.Clave AND P.Numero=E.Numero AND P.Denominacion='Mexico sin ti no
estamos completos'
```

	Clave	Descripcion
1	1030	Vanilla 4/33
2	1230	Cemento
3	1430	Pintura B1022

Número de Tuplas: 3

- 2) Los materiales (clave y descripción) que han sido proporcionados por el proveedor "Acme tools".

```
SELECT E.Clave, M.Descripcion
FROM Materiales as M, Entregan as E, Proveedores as P
WHERE M.Clave=E.Clave AND P.RFC=E.RFC AND P.RazonSocial='Acme tools'
```

	Clave	Descripcion
--	-------	-------------

Número de Tuplas: 0

- 3) El RFC de los proveedores que durante el 2000 entregaron en promedio cuando menos 300 materiales.

```
SELECT E.RFC
FROM Entregan as E
WHERE E.Cantidad>=300 AND (Fecha Between '01-JAN-2000' AND '31-DEC-2000')
```

	RFC
1	BBBB800101
2	BBBB800101
3	EEEE800101

Número de Tuplas: 18

- 4) El Total entregado por cada material en el año 2000.

```
SELECT E.Clave, SUM(Cantidad) as 'Total'
FROM Entregan as E
WHERE (Fecha Between '01-JAN-2000' AND '31-DEC-2000')
GROUP BY E.Clave
```

	Clave	Total
1	1000	7.00
2	1010	1195.00
3	1030	295.00

Número de Tuplas: 22

- 5) La Clave del material más vendido durante el 2001. (se recomienda usar una vista intermedia para su solución)

```
SELECT TOP 1 E.Clave, SUM(Cantidad) as 'Total'
FROM Entregan as E
WHERE (Fecha Between '01-JAN-2001' AND '31-DEC-2001')
GROUP BY E.Clave
ORDER BY Total DESC
```

	Clave	Total
1	1020	1060.00

Número de Tuplas: 1

- 6) Productos que contienen el patrón 'ub' en su nombre.

```
SELECT descripcion
FROM Materiales AS M
WHERE descripcion LIKE '%ub%';
```

	descripcion
1	Recubrimiento P1001
2	Recubrimiento P1010
3	Recubrimiento P1019

Número de Tuplas: 12

- 7) Denominación y suma del total a pagar para todos los proyectos.

```
SELECT P.Denominacion, SUM(E.Cantidad*(M.Costo*(PorcentajeImpuesto/100)+1)) as "Total"
FROM Entregan AS E, Proyectos AS P, Materiales AS M
WHERE P.Numero=E.Numero AND M.Clave=E.Clave
GROUP BY P.Denominacion
```

	Denominacion	Total
1	Ampliación de la carretera a la huasteca	20204.1940000000
2	Aztecón	4777.2190000000
3	CIT Campeche	5057.0820000000

Número de Tuplas: 20

- 8) Denominación, RFC y RazonSocial de los proveedores que se suministran materiales al proyecto Televisa en acción que no se encuentran apoyando al proyecto Educando en Coahuila (Solo usando vistas).

```
CREATE VIEW televisaNoCoahuila1 AS
(SELECT DISTINCT P.Denominacion, PR.RFC, PR.RazonSocial
FROM Entregan AS E, Proyectos AS P, Proveedores as PR
WHERE P.Numero=E.Numero AND PR.RFC=E.RFC AND P.Denominacion LIKE 'Tel%' AND PR.RFC
NOT IN (SELECT PR.RFC FROM Entregan AS E, Proyectos AS P, Proveedores as PR
WHERE P.Numero=E.Numero AND PR.RFC=E.RFC AND P.Denominacion LIKE 'Edu%' ))
select * from televisaNoCoahuila1
DROP VIEW televisaNoCoahuila1
```

	Denominacion	RFC	RazonSocial
1	Televisa en acción	CCCC800101	La Ferra
2	Televisa en acción	DDDD800101	Cecoferra

Número de Tuplas: 2

- 9) Denominación, RFC y RazonSocial de los proveedores que se suministran materiales al proyecto Televisa en acción que no se encuentran apoyando al proyecto Educando en Coahuila (Sin usar vistas, utiliza not in, in o exists).

```
SELECT DISTINCT P.Denominacion, PR.RFC, PR.RazonSocial
FROM Entregan AS E, Proyectos AS P, Proveedores as PR
WHERE P.Numero=E.Numero AND PR.RFC=E.RFC AND P.Denominacion LIKE 'Tel%' AND PR.RFC
NOT IN (SELECT PR.RFC FROM Entregan AS E, Proyectos AS P, Proveedores AS PR
WHERE P.Numero=E.Numero AND PR.RFC=E.RFC AND P.Denominacion LIKE 'Edu%')
```

	Denominacion	RFC	RazonSocial
1	Televisa en acción	CCCC800101	La Ferra
2	Televisa en acción	DDDD800101	Cecoferra

Número de Tuplas: 2

- 10) Costo de los materiales y los Materiales que son entregados al proyecto Televisa en acción cuyos proveedores también suministran materiales al proyecto Educando en Coahuila.

```
SELECT M.Costo, M.Descripcion
FROM Entregan AS E, Proyectos AS P, Materiales AS M, Proveedores AS PR
WHERE P.Numero=E.Numero AND PR.RFC=E.RFC AND M.Clave=E.Clave AND M.Clave=E.Clave
AND P.Denominacion LIKE 'Tel%' AND PR.RFC
IN(
SELECT PR.RFC
FROM Entregan AS E, Proyectos AS P, Proveedores as PR
WHERE P.Numero=E.Numero AND PR.RFC=E.RFC AND P.Denominacion LIKE 'Edu%')
```

	Costo	Descripcion
1	50.00	Ladrillos rojos
2	34.00	Tepetate

Número de Tuplas: 2

- 11) Nombre del material, cantidad de veces entregados y total del costo de dichas entregas por material de todos los proyectos.

```
SELECT M.Descripcion, P.Denominacion, SUM(E.Cantidad) as 'cantidadEntrega',
SUM(E.Cantidad*(M.Costo*(PorcentajeImpuesto/100)+1)) as "costoTotal"
FROM Entregan AS E, Materiales AS M, Proyectos AS P
WHERE M.Clave=E.Clave AND P.Numero=E.Numero
GROUP BY M.Descripcion, P.Denominacion
```

	Descripcion	Denominacion	cantidadEntrega	costoTotal
1	Cantera rosa	Ampliación de la carretera a la huasteca	162.00	5498.9280000000
2	Pintura C1010	Ampliación de la carretera a la huasteca	364.00	1601.6000000000
3	Recubrimiento P1010	Ampliación de la carretera a la huasteca	978.00	6098.8080000000

Número de Tuplas: 87

INVESTIGACIÓN

	Oracle	SQL Server	MySql	Mongo DB
Requerimientos Generales de Hardware	<p>Windows</p> <ul style="list-style-type: none"> • AMD Opteron, Intel Pentium® at 500 MHz or faster, or Intel EM64T • Minimum 500 MB free disk space for installation, 10 GB recommended • Minimum 1 GB physical memory, 4 GB recommended <p>Solaris</p> <ul style="list-style-type: none"> • AMD Opteron, Intel Pentium® at 500 MHz or faster, or Intel EM64T • Minimum 500 MB free disk space for installation, 10 GB recommended • Minimum 1 GB physical memory, 4 GB recommended <p>Linux</p> <ul style="list-style-type: none"> • AMD Opteron, Intel Pentium® at 500 MHz or faster, or Intel EM64T • Minimum 500 MB free disk space for installation, 10 GB recommended • Minimum 1 GB physical memory, 4 GB recommended 	<p>Express Editions: 512 MB</p> <p>All other editions: 1 GB</p> <p>x64 Processor: 1.4 GHz</p> <p>6 GB of available hard-disk space.</p>	<p>No hay límite en el número de CPU, con todas las funciones, soporte para varios sistemas operativos</p>	<p>Para servidores:</p> <p>>400 hosts monitoreados:</p> <p>4 GB RAM</p> <p>200 GB HDD</p> <p>4+ Cores</p> <p>>200 hosts monitoreados:</p> <p>15 GB RAM</p> <p>500 GB HDD</p> <p>8+ Cores</p>

Ambientes o Plataformas en las que pueden operar	AIX HP-UX Linux OS X Solaris Windows z/OS	Linux Windows	FreeBSD Linux OS X Solaris Windows	Linux OS X Solaris Windows
Costos de Implementación y Mantenimiento	Versión gratuita Oracle Express Licencia Premium SE \$17400 EE \$47500	Posee una versión libre: SQL Server Express Y una de pago: SQL Server La licencia para la versión comercial varía entre \$ 600 y \$ 10,000 dólares	Gratuito debido a que se distribuye bajo licencia GPL. Para soporte por suscripción: de 600 a 6000 dólares.	Gratuito y sólo es de pago cuando se requiere soporte
Ventajas	Es muy confiable puede trabajar con enormes tamaños de bases de datos y cargas de trabajo de aplicaciones masivas. Es portable y está disponible en diferentes plataformas de SO, incluida la misma base de código y el mismo conjunto de documentación. Es la más rápida. Soporta el lenguaje de manipulación de datos PL/SQL que permite implementar diseños “activos”, con triggers y procedimientos almacenados, con una integridad referencial declarativa bastante potente.	Viene con todas las herramientas SQL estándar Soporte de correo electrónico. En su versión de pago el límite de las bases de datos viene dado por el SO	Gratuito Multiplataforma Más rápido que sus competidores Open Source El tamaño de las bases de datos viene dado por el SO Soporta transacciones	Open source Empleo de estructuras BSON en lugar de tablas Esquema dinámico

	<p>Permite el uso de peticiones para la mejora de la eficiencia, de replicación e incluso ciertas versiones admiten la administración de bases de datos distribuidas.</p> <p>Orientación hacia INTERNET.</p>		<p>Permite replicación</p> <p>Altamente escalable, brinda soporte para la mayoría de los tipos de índices y métodos de almacenamiento.</p>	
Desventajas	<p>Coste de licencia elevado.</p> <p>Configuración del lado del servidor demasiado compleja, y de realizarse mal nos ralentizará nuestros sistemas.</p> <p>Interfaz de usuario débil, proceso complejo de instalación y mantenimiento.</p> <p>Curva de aprendizaje un poco lenta.</p>	<p>No permite replicación</p> <p>En la versión gratuita el límite de las bases de datos es de 10 GB.</p> <p>Características limitadas, pero escala fácilmente a la versión Enterprise.</p>	<p>No posee servicio de reportes</p> <p>La compatibilidad con ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) depende del motor de almacenamiento</p> <p>Difícil de escalar</p>	<p>De momento es una tecnología muy joven, es decir, necesita tiempo para volverse robusta en el empleo de soluciones</p> <p>No es adecuada para transacciones complejas</p> <p>No tiene un reemplazo para las soluciones de herencia</p>
Porcentaje del Mercado que Controlan	3.67%	13.25%	33.08%	4.64%

REFERENCIAS:

<https://academy.datastax.com/planet-cassandra/what-is-nosql>

<http://www.dadbm.com/what-is-oracle-database-rdbms/>

<https://docs.oracle.com/en/database/oracle/oracle-database/12.2/ntdbi/oracle-database-minimum-hardware-requirements.html#GUID-B7DE91F4-FD3D-4E7B-96C2-F9D2C60B27D2>

<https://www.codecademy.com/articles/what-is-rdbms-sql>

https://docs.oracle.com/cd/E24191_01/common/install/system_requirements.html

<https://guiadev.com/mysql-vs-mongodb/>

<https://www.oracle.com/assets/technology-price-list-070617.pdf>

<https://docs.microsoft.com/en-us/sql/sql-server/install/hardware-and-software-requirements-for-installing-sql-server?view=sql-server-2017>

<https://db-engines.com/en/system/Oracle>

<https://db-engines.com/en/system/Microsoft+SQL+Server>

<https://db-engines.com/en/system/MySQL>

<https://db-engines.com/en/system/MongoDB>

<https://www.datanyze.com/market-share/databases>

<https://docs.opsmanager.mongodb.com/current/core/requirements/>