

Investigación sobre DBMS empresariales

ORACLE**-Requerimientos Generales de Hardware**

1-2 GB para las tareas de evaluación

Un mínimo de 4 GB para los servidores de producción

- Ambientes o plataformas en las que pueden operar.

Multiplataforma: puede ejecutarse desde un PC hasta una supercomputadora.

- Costos de implementación y mantenimiento.

Licencia de Oracle Database	Unit Price
Database Standard Edition	\$5.7312
Database Enterprise Edition	\$6.1613
Database Enterprise Edition High Performance	\$7.0753
Database Enterprise Edition Extreme Performance	\$7.9892

- Ventajas y desventajas de su uso.**Ventajas:**

- Herramienta de administración gráfica intuitiva y cómoda de utilizar
- Control de acceso: tecnologías avanzadas para vigilar la entrada a los datos.
- Protección de datos: seguridad completa en el entorno de producción y de pruebas y gestión de copias de seguridad.
- Permite el uso de particiones para la mejora de la eficiencia, de replicación e incluso ciertas versiones admiten la administración de bases de datos distribuidas.

Desventajas:

- Precio elevado
- Configuración compleja
- Cote de info elevado

- Porcentaje del mercado que controlan

24% del mercado

Ligas de consulta:

<https://docs.oracle.com/cd/E19656-01/821-1717/hardware/index.html>

<https://www.oracle.com/mx/it-infrastructure/>

<https://www.cursosfemxa.es/blog/5-grandes-ventajas-de-oracle>

<http://oraclebddepn.blogspot.com/2013/05/ventajas-y-desventajas.html>

<http://ict.udlap.mx/people/carlos/is341/bases06.html>

SQL Server**- Requerimientos Generales de Hardware**

Ediciones Express: 512 MB

Las demás ediciones: 1 GB

Al menos 4 GB, que debe aumentar a medida que el tamaño de la base de datos aumente para asegurar un rendimiento óptimo

- Ambientes o plataformas en las que pueden operar.

Compatible con las versiones de Windows

- Costos de implementación y mantenimiento.

Ediciones de SQL Server 2017	Modelo de licencias	Precio de Open No Level (USD)
Enterprise	Por núcleo	\$14,256
Standard, por núcleo	Por núcleo	\$3,717
Standard, Server + CAL	Server + CAL	\$931
Desarrollador	Por usuario	Gratis

- Ventajas y desventajas de su uso.

Ventajas:

- Es útil para manejar y obtener datos de la red de redes.
- Nos permite olvidarnos de los ficheros que forman la base de datos
- SQL permite administrar permisos a todos.

Desventajas:

- La relación, calidad y el precio está muy debajo comparado con oracle.
- Tiene muchos bloqueos a nivel de página, un tamaño de página fijo y demasiado pequeño, una pésima implementación de los tipos de datos variables.
- Utiliza mucho la memoria RAM para las instalaciones y utilización de software.

- Porcentaje del mercado que controlan

66,5 del mercado de habla hispana

Ligas:

<https://docs.microsoft.com/es-es/sql/sql-server/install/hardware-and-software-requirements-for-installing-sql-server?view=sql-server-ver15#:~:text=%20La%20memoria%20m%C3%ADnima%20necesaria%20para,memoria%20m%C3%ADnima%20de%20SQL%20Server%20>.

<https://sqlserver4b.weebly.com/ventajas-y-desventajas.html>

<https://www.dealerworld.es/mercado-en-cifras/sql-server-lidera-las-ventas-de-bases-de-datos-sobre-nt>

MySql**- Requerimientos Generales de Hardware**

512 Mb de memoria Ram.

1024 Mb maquina virtual.

1 GB de espacio de disco duro.

Sistema operativo:Windows, Linux y Unix.

Arquitectura del sistema 32/64 bit.

Protocolo de red TCP/IP.

- Ambientes o plataformas en las que pueden operar.

- AIX.
- BSD.
- FreeBSD.
- HP-UX.
- GNU/Linux.
- Mac OS X.
- NetBSD.
- Novell Netware.

- Costos de implementación y mantenimiento.

Aplicación	Coste mensual
Instancia de prueba	9 USD
Instancia de prueba que se detiene cuando no se usa	3 USD
Base de datos de producción de alto rendimiento	7670 USD

- Ventajas y desventajas de su uso.

Ventajas:

- MySQL software es Open Source
- Facilidad de configuración e instalación.
- Bajo costo en requerimientos para la elaboración de bases de datos
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.

Desventajas:

- No es intuitivo, como otros programas (ACCESS).
- Un gran porcentaje de las utilidades de MySQL no están documentadas.

- Porcentaje del mercado que controlan.

44% del mercado

Ligas:

<http://ict.udlap.mx/people/carlos/is341/bases06.html>

<https://mysqldaniel.wordpress.com/ventajas-y-desventajas/>

Redis

- Requerimientos Generales de Hardware.

An empty instance uses ~ 3MB of memory.

1 Million small Keys -> String Value pairs use ~ 85MB of memory.

1 Million Keys -> Hash value, representing an object with 5 fields, use ~ 160 MB of memory.

- Ambientes o plataformas en las que pueden operar.

Unix, Linux y sus derivados, Solaris, OS/X

- Costos de implementación y mantenimiento.

Nombre de caché	Tamaño de caché	Precio
C0	250 MB	€0,019/hora
C1	1 GB	€0,047/hora
C2	2,5 GB	€0,076/hora
C3	6 GB	€0,152/hora

- Ventajas y desventajas de su uso.

Ventajas:

- Una velocidad muy por encima de la media respecto a otras DB de su tipo, gracias a su almacenamiento en memoria
- Posibilidad de persistir datos en disco para recuperación ante fallas
- Fácil configuración
- Alta disponibilidad

Desventajas:

- El método de persistencia RDB consume mucho I/O (escritura en disco)
- Todos los datos trabajados deben encajar en la memoria (en caso de no usar persistencia física)

Ligas:

<https://blog.bi-geek.com/redis-para-principiantes/>

<https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>

Consultar:**select * from materiales**

Salida:

44 rows

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

select * from materiales**where clave=1000**

Salida:

1 row

	Clave	Descripcion	Costo
1	1000	Varilla 3/16	100.00

select clave,rfc,fecha from entregan

Salida:

132 rows

	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

select * from materiales,entregan**where materiales.clave = entregan.clave**

Salida:

132 rows

	Clave	Descripcion	Costo	Clave	RFC	Numero	Fecha
1	1000	Varilla 3/16	100.00	1000	AAAA800101	5000	1998-07-08 00:00:0
2	1000	Varilla 3/16	100.00	1000	AAAA800101	5019	1999-08-08 00:00:0
3	1000	Varilla 3/16	100.00	1000	AAAA800101	5019	2000-04-06 00:00:0

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

No, no aparecería

select * from entregan,proyectos**where entregan.numero < = proyectos.numero**

Salida:

1188 rows

	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

(select * from entregan where clave=1450)

union

(select * from entregan where clave=1300)

Salida:

3 rows

	Clave	RFC	Numero	Fecha	Cantidad
1	1300	GGGG800101	5005	2002-06-10 00:00:00.000	521.00
2	1300	GGGG800101	5005	2003-02-02 00:00:00.000	457.00
3	1300	GGGG800101	5010	2003-01-08 00:00:00.000	119.00

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

Compruébalo.

select * from entregan where clave=1450 or clave=1300

Salida:

3 rows

	Clave	RFC	Numero	Fecha	Cantidad
1	1300	GGGG800101	5005	2002-06-10 00:00:00.000	521.00
2	1300	GGGG800101	5005	2003-02-02 00:00:00.000	457.00
3	1300	GGGG800101	5010	2003-01-08 00:00:00.000	119.00

(select clave from entregan where numero=5001)

intersect

(select clave from entregan where numero=5018)

Salida:

1 row

	clave
1	1010

select * from entregan where clave=1000

Salida:

3 rows

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

select * from entregan,materiales

Salida:

5808 rows

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

¿Cómo está definido el número de tuplas de este resultado en términos del número de tuplas de entregan y de materiales?

Es como una multiplicación de cada una de la tuplas de materiales, por cada una de las de entregan.

Plantea ahora una consulta para obtener las descripciones de los materiales entregados en el año 2000.

Select Descripcion

From Materiales, Entregan

Where materiales.clave = entregan.clave and Entregan.Fecha >= '01-01-2000' and

Entregan.Fecha <= '31-12-2000'

Salida:

28 rows

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

¿Por qué aparecen varias veces algunas descripciones de material?

Porque fueron hechas varias entregas de ese material

Select Distinct Descripcion

From Materiales, Entregan

Where materiales.clave = entregan.clave and Entregan.Fecha >= '01-01-2000' and

Entregan.Fecha <= '31-12-2000'

Salida:

22 rows

	Descripcion
1	Arena
2	Block
3	Cantera rosa

¿Qué resultado obtienes en esta ocasión?

Las row que se repetían ya no se repiten

Obtén los números y denominaciones de los proyectos con las fechas y cantidades de sus entregas, ordenadas por número de proyecto, presentando las fechas de la más reciente a la más antigua.

Select P.numero, P.Denominacion, E.Fecha,E.cantidad

FROM Proyectos P, Entregan E

Where E.Numero = P.Numero

ORDER BY P.Numero, E.Fecha DESC

Salida:

132 rows

	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

SELECT * FROM materiales where Descripcion LIKE 'Si%'

¿Qué resultado obtienes?

Salida:

2 rows

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

Explica que hace el símbolo '%'.

Indica que después hay caracteres

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

¿Qué resultado obtienes?

Conjunto vacío

Explica a qué se debe este comportamiento.

Solo toma en cuenta Si y como no hay ninguno así, no devuelve nada

```
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 siguiente código?

¿Qué resultado obtienes? ¿¿¿??

¿Para qué sirve DECLARE?

Declarar variables

¿Cuál es la función de @foo?

Es una variable que guarda una cadena de caracteres

¿Que realiza el operador SET?

activa varias opciones que afectan al funcionamiento del servidor y el cliente.

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

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


Muestra los RFC de entregan que empiecen desde A hasta D

Salida:

72 rows

	RFC
1	AAAA800101
2	AAAA800101
3	AAAA800101

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

Muestra todos los RFC que no empiecen con A

Salida:

114 rows

	RFC
1	BBBB800101
2	BBBB800101
3	BBBB800101

SELECT Numero FROM Entregan WHERE Numero LIKE '___6' ;

Muestra todos los números de los proyectos que después de tres caracteres tengan un 6

Salida:

14 rows

	Numero
1	5016
2	5016
3	5006

¿Cómo filtrarías rangos de fechas?

Select Distinct Descripcion

From Materiales M, Entregan E

Where M.clave = E.clave and E.Fecha Between '01-01-2000' and '31-12-2000'

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])

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

Es la donde empieza la subconsulta

Tomando de base la consulta anterior del EXISTS, realiza el query que devuelva el mismo resultado, pero usando el operador IN

Select RFC, Cantidad, Fecha, numero

From Entregan E

where E.RFC In (select RFC from Proveedores where RazonSocial Like 'La%' and Numero Between 5000 and 5010)

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 E.RFC, Cantidad, Fecha, numero

From Entregan E, Proveedores P

where E.RFC = P.RFC AND E.RFC NOT IN (select E.RFC from Proveedores where Numero < 5000 Union select RFC from Proveedores where numero > 5010) AND P.RazonSocial LIKE 'La%'

Select E.Cantidad

From Entregan E

where clave =Any (select M.Clave from Materiales M where M.Descripcion = 'Block')

¿Qué hace la siguiente sentencia? Explica por qué.

SELECT TOP 2 * FROM Proyectos

Selecciona las dos primeras rows registradas de proyectos

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

SELECT TOP Numero FROM Proyectos

No se pudo completar la consulta, porque no sabe cuántas filas tomar del top

Comprueba lo anterior, creando vistas para cinco de las consultas que planteaste anteriormente en la práctica . Posteriormente revisa cada vista creada para comprobar que devuelve el mismo resultado.

Vistas:

----1

Create view EntreganTodo

as select clave,rfc,fecha from entregan

---2

Create view Materiales1000

as select * from materiales where clave=1000

---3

Create view descripanio200

as Select Distinct Descripcion

From Materiales M, Entregan E

Where M.clave = E.clave and E.Fecha Between '01-01-2000' and '31-12-2000'

--4

Select E.Cantidad

From Entregan E

where clave =Any (select M.Clave from Materiales M where M.Descripcion = 'Block')

--5

Create view DescMateriales

as SELECT * FROM materiales where Descripcion LIKE 'Si%'

En el reporte incluye la sentencia, una muestra de la salida (dos o tres renglones) y el número de renglones que SQL Server reporta al final de la consulta.

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

SELECT M.Clave, M.Descripcion FROM Materiales M, Entregan E, Proyectos P

Where M.Clave = E.clave And E.Numero= P.Numero and E.Numero IN(Select numero from proyectos p where p.Denominacion like 'Me%')

Salida:

3 rows

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

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

SELECT M.Clave, M.Descripcion FROM Materiales M, Entregan E, Proyectos P

Where M.Clave = E.clave And E.RFC= P.Numero and E.RFC IN(Select RFC from proveedores p where p.RazonSocial like 'ac%')

Salida:

0 rows

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

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

SELECT RFC, Sum(Cantidad) as sum FROM Entregan E

Where E.Fecha Between '01-01-2000' and '31-12-2000'

GROUP BY RFC

Having sum(Cantidad) >=300

Salida:

8 rows

	RFC	Sum
1	AAAA800101	948.00
2	BBBB800101	1962.00
3	CCCC800101	744.00
4	DDDD800101	943.00
5	EEEE800101	2656.00
6	FFFF800101	1122.00
7	GGGG800101	1063.00
8	HHHH800101	979.00

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

```
SELECT Clave, Sum(E.Cantidad) as 'Total'
FROM Entregan E
Where E.Fecha Between '01-01-2000' and '31-12-2000'
GROUP BY Clave
```

Salida:

22 rows

	Clave	Total
1	1000	7.00
2	1010	1195.00
3	1030	295.00

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

Create View Vendido2001

```
as select Clave, sum(cantidad) as 'Ventas' from Entregan E where E.fecha Between
'01-01-2001' and '31-12-2001' group by Clave ;
```

```
select Clave
From Vendido2001
Having MAX(Ventas);
-- :o
```

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

```
Select * From Materiales Where Descripcion Like '%ub%'
```

Salida:

12 rows

	Clave	Descripcion	Costo	PorcentajImpuesto
1	1180	Recubrimiento P1001	200.00	2.36
2	1190	Recubrimiento P1010	220.00	2.38
3	1200	Recubrimiento P1019	240.00	2.40

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

```
Select Denominacion, Sum(E.Cantidad) as 'totalpagar' From Proyectos p, Entregan E Where
E.numero = p.numero
```

--:o

