

## **ORACLE**

### Requerimientos mínimos:

- Memoria: entre 512MB y 1GB
- Espacio de disco duro: Casi todos 250MB
- JVC: J2SE 5.0 y Java SE 6

### Ambientes en lo que puede operar:

- OpenSolaris
- En Macintosh OS 10.4, 10.5 (Intel, Power)
- Windows 2000 SP4+
- AIX 5.2, 5.3, 6.1
- Ubuntu Linux 8.04, versión Hardy
- Red Hat Enterprise Linux 5.x de 64 bits
- Entre Otros

### Costos de implementación y Mantenimiento:

Alrededor de 5000 dólares.

### Ventajas y desventajas de uso:

- Ventajas:
  - Reduce significativamente la cantidad para aprovisionar una nueva base de datos
  - Todas las tareas de rutina (instalación / parcheo / copia de seguridad / escalado) son manejadas por DB-as-a-service.
  - Marco de seguridad incorporado que evita el acceso no autorizado.
  - Proporcione DR y alta disponibilidad listos para usar.
- Desventajas:
  - No hay acceso al host físico de la base de datos. Esto se espera de una base de datos administrada. Todo debe hacerse a través de la consola.
  - Debido a la falta de acceso físico al host, ciertas funciones no son compatibles, como los espacios de tabla transportables.
  - Ciertos tipos de datos especiales (como XMLType) no están permitidos.

### Porcentaje del Mercado que controlan:

2% del Mercado de servidores similares a ORACLE.

### Referencias:

<https://docs.oracle.com/cd/E19879-01/821-1040/abpai/index.html>

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

<https://www.trustradius.com/products/oracle-autonomous-database/reviews?qs=pros-and-cons>

<https://www.datanyze.com/market-share/databases--272/oracle-database-market-share>

## **SQL Server 2012 Express con Tools Edition**

### Requerimientos mínimos:

- Procesador: Intel o compatible a 2GHz
- Memoria: 4GB
- Espacio de disco duro: 4GB
- Complementos:
  - Navegador: Microsoft Internet Explorer 10
  - NET Framework 3.5 SP1 ó NET 4
  - Windows PowerShell 1.0

### Ambientes en lo que puede operar:

- Windows Server 2008 R2
- Windows Server 2008 c/SP2

- Windows 7 Professional c/SP1

Costos de implementación y Mantenimiento:

Alrededor de 800 dólares.

Ventajas y desventajas de uso:

- Ventajas:
  - Procesamiento de consultas más rápido
  - Sin necesidad de saber de programación
  - Lenguaje estandarizado
  - Portable
  - Lenguaje interactivo
- Desventajas:
  - Interface compleja
  - Control parcial del sistema

Porcentaje del Mercado que controlan:

18.01% del Mercado de servidores similares a SQL

### Referencias:

[https://conocimiento.blob.core.windows.net/conocimiento/MicroSitio/Manual Instalacion CONTPAQI.pdf](https://conocimiento.blob.core.windows.net/conocimiento/MicroSitio/Manual%20Instalacion%20CONTPAQI.pdf)

<https://www.geeksforgeeks.org/advantages-and-disadvantages-of-sql/>

### MYSQL

Requerimientos mínimos:

- Procesador: 64bit x86
- Memoria: 4GB
- Monitor: 1024×768

Ambientes en lo que puede operar:

- Ubuntu 21.04
- Windows Server 2019
- Windows 10
- macOS 11

Costos de implementación y Mantenimiento:

Alrededor de 5000 dólares.

Ventajas y desventajas de uso:

- Ventajas:
  - Excelente soporte para copias de seguridad y replicación de datos
  - Open-Source software
  - Buena seguridad de datos
  - Fácil escalabilidad
  - Buen manejo de datos
  - Simple de usar
- Desventajas:
  - Sin funciones de informes y análisis
  - Sin soporte para grandes bases de datos
  - Falta de compatibilidad con estructura de tablas comunes
  - Falta de replicación de datos de flujo

Porcentaje del Mercado que controlan:

14.78% del Mercado similares a MYSQL

### Referencias:

<https://www.mysql.com/support/supportedplatforms/workbench.html>

<https://www.mysql.com/tcosavings/>

## MongoDB

Requerimientos mínimos:

- Memoria: 1GB por 100.000 assets.
- Memoria de disco duro: Espacio de disco duro SDD = assets \* 0.01

Ambientes en lo que puede operar:

- Casi todos los sistemas actuales.

Costos de implementación y Mantenimiento:

- \$10/500 hours of request runtime Excluding Sync
- \$0.08/1M minutes of Sync runtime
- \$2.00/1M application requests

Ventajas y desventajas de uso:

- Ventajas:
  - Buen desempeño
  - Simplicidad de uso
  - Flexibilidad
  - Documentación precisa
- Desventajas:
  - Transacciones
  - Joins muy difíciles
  - Datos duplicados son difícil de manejar

Porcentaje del Mercado que controlan:

3.74% del Mercado de servidores similares a MongoDB

## Referencias:

[https://learn.fotoware.com/On-Premises/FotoWeb/05\\_Configuring\\_sites/Setting\\_the\\_MongoDB\\_instance\\_that\\_FotoWeb\\_uses/MongoDB\\_disk\\_and\\_memory\\_requirements](https://learn.fotoware.com/On-Premises/FotoWeb/05_Configuring_sites/Setting_the_MongoDB_instance_that_FotoWeb_uses/MongoDB_disk_and_memory_requirements)

<https://docs.mongodb.com/manual/administration/production-notes/>

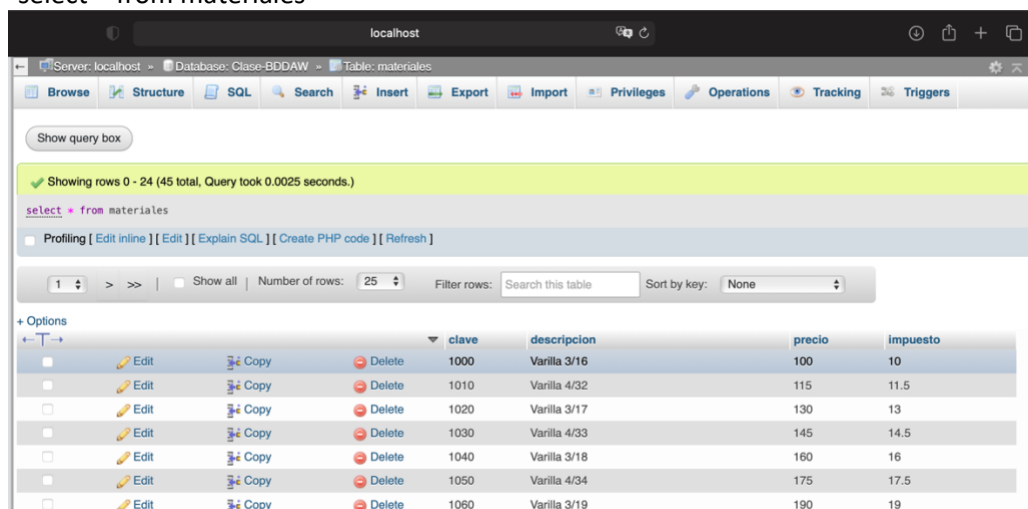
<https://www.mongodb.com/pricing>

<https://www.virtual-dba.com/blog/pros-and-cons-of-mongodb/>

## Consultas Básicas en SQL

Consulta de una tabla completa:

`select * from materiales`



The screenshot shows a web-based SQL client interface. At the top, there's a navigation bar with tabs like 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers'. Below this, a 'Show query box' button is visible. The main area displays the results of a query: 'select \* from materiales'. A status bar indicates 'Showing rows 0 - 24 (45 total, Query took 0.0025 seconds.)'. Below the query, there are links for 'Profiling', 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. A toolbar shows 'Show all', 'Number of rows: 25', 'Filter rows: Search this table', and 'Sort by key: None'. The results are shown in a table with columns: 'clave', 'descripcion', 'precio', and 'impuesto'. The table contains 8 rows of data.

	clave	descripcion	precio	impuesto
<input type="checkbox"/>	1000	Vanilla 3/16	100	10
<input type="checkbox"/>	1010	Vanilla 4/32	115	11.5
<input type="checkbox"/>	1020	Vanilla 3/17	130	13
<input type="checkbox"/>	1030	Vanilla 4/33	145	14.5
<input type="checkbox"/>	1040	Vanilla 3/18	160	16
<input type="checkbox"/>	1050	Vanilla 4/34	175	17.5
<input type="checkbox"/>	1060	Vanilla 3/19	190	19

### Selección:

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

Showing rows 0 - 0 (1 total, Query took 0.0009 seconds.)

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

Options: Show all | Number of rows: 25 | Filter rows: Search this table

	clave	descripcion	precio	impuesto
<input type="checkbox"/>	1000	Varilla 3/16	100	10

Options: Show all | Number of rows: 25 | Filter rows: Search this table

### Proyección:

```
select clave,rfc,fecha from entregan
```

Showing rows 0 - 24 (87 total, Query took 0.0017 seconds.)

```
select clave,rfc,fecha from entregan
```

Options: Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

	clave	rfc	fecha
<input type="checkbox"/>	1000	AAAA800101	2001-12-13
<input type="checkbox"/>	1000	AAAA800101	1999-07-13
<input type="checkbox"/>	1010	BBBB800101	1998-07-28
<input type="checkbox"/>	1010	BBBB800101	1997-02-09
<input type="checkbox"/>	1020	CCCC800101	2003-12-16
<input type="checkbox"/>	1020	CCCC800101	2000-03-29

### Reunión Natural:

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

Showing rows 0 - 24 (87 total, Query took 0.0024 seconds.)

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

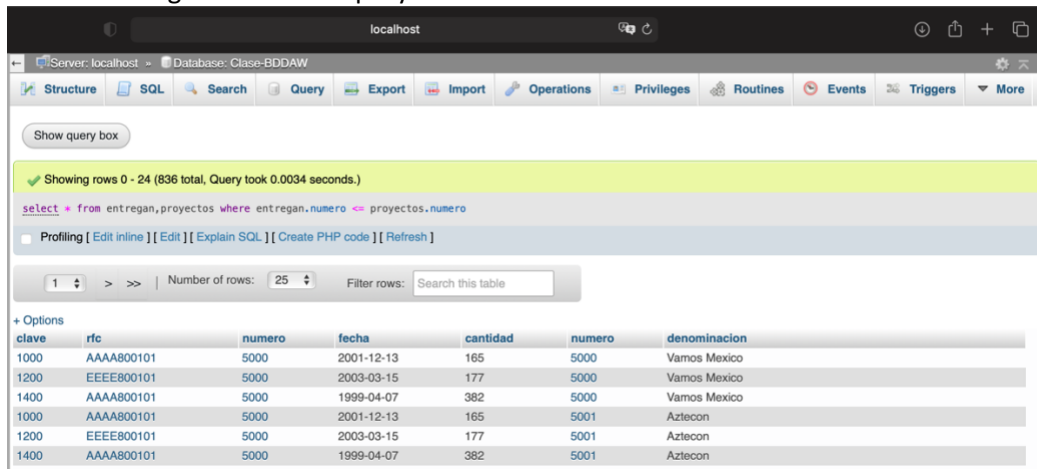
Options: Show all | Number of rows: 25 | Filter rows: Search this table

clave	descripcion	precio	impuesto	clave	rfc	numero	fecha	cantidad
1000	Varilla 3/16	100	10	1000	AAAA800101	5000	2001-12-13	165
1000	Varilla 3/16	100	10	1000	AAAA800101	5019	1999-07-13	254
1010	Varilla 4/32	115	11.5	1010	BBBB800101	5001	1998-07-28	528
1010	Varilla 4/32	115	11.5	1010	BBBB800101	5018	1997-02-09	523
1020	Varilla 3/17	130	13	1020	CCCC800101	5002	2003-12-16	582
1020	Varilla 3/17	130	13	1020	CCCC800101	5017	2000-03-29	8

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

### Reunión con criterio específico:

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

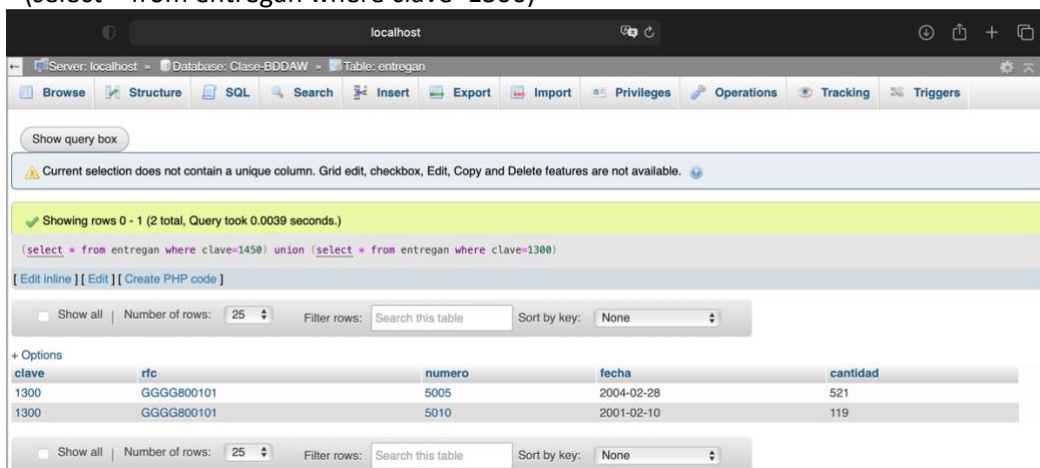


The screenshot shows the phpMyAdmin interface with a query executed: `select * from entregan,proyectos where entregan.numero <= proyectos.numero`. The result shows 24 rows. The table structure is as follows:

clave	rfc	numero	fecha	cantidad	numero	denominacion
1000	AAAA800101	5000	2001-12-13	165	5000	Vamos Mexico
1200	EEEE800101	5000	2003-03-15	177	5000	Vamos Mexico
1400	AAAA800101	5000	1999-04-07	382	5000	Vamos Mexico
1000	AAAA800101	5000	2001-12-13	165	5001	Aztlecon
1200	EEEE800101	5000	2003-03-15	177	5001	Aztlecon
1400	AAAA800101	5000	1999-04-07	382	5001	Aztlecon

### Unión:

```
(select * from entregan where clave=1450)
union
(select * from entregan where clave=1300)
```

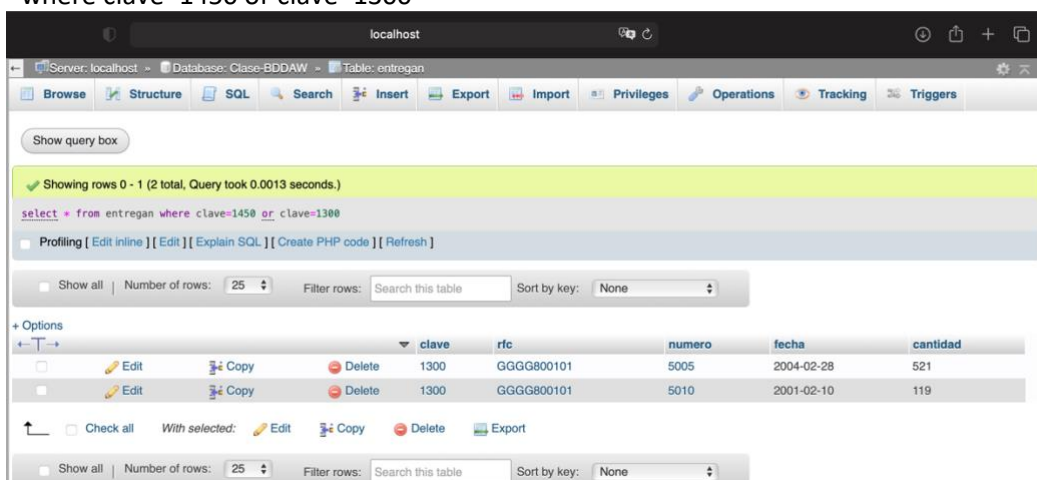


The screenshot shows the phpMyAdmin interface with a UNION query executed: `(select * from entregan where clave=1450) union (select * from entregan where clave=1300)`. The result shows 2 rows. The table structure is as follows:

clave	rfc	numero	fecha	cantidad
1300	GGGG800101	5005	2004-02-28	521
1300	GGGG800101	5010	2001-02-10	119

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

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



The screenshot shows the phpMyAdmin interface with a query executed: `select * from entregan where clave=1450 or clave=1300`. The result shows 2 rows. The table structure is as follows:

clave	rfc	numero	fecha	cantidad
1300	GGGG800101	5005	2004-02-28	521
1300	GGGG800101	5010	2001-02-10	119

### Intersección:

```
(select clave from entregan where numero=5001)
intersect
(select clave from entregan where numero=5018) PARA ORACLE
```

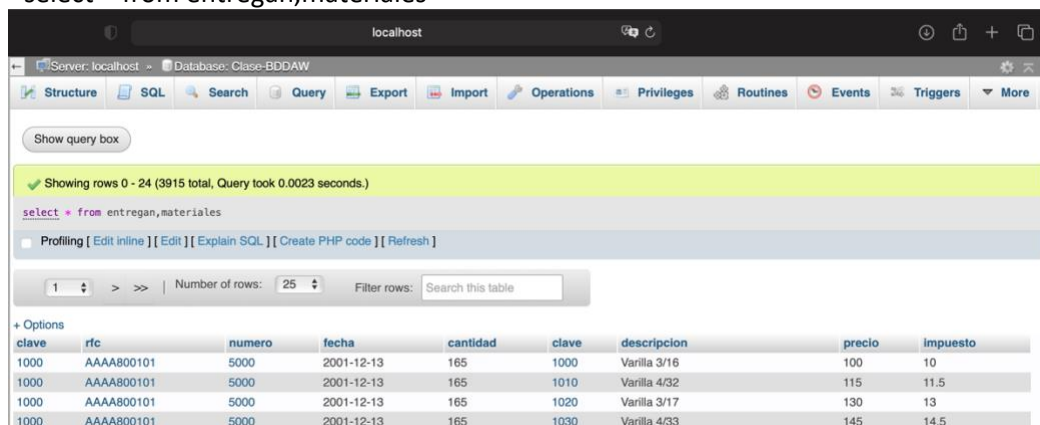
### Diferencia:

```
(select * from entregan)
minus
(select * from entregan where clave=1000)
```

La palabra "minus" es una palabra reservada que no está definida en SQL Server, define una consulta que regrese el mismo resultado.

### Producto cartesiano:

```
select * from entregan,materiales
```



clave	rfc	numero	fecha	cantidad	clave	descripcion	precio	impuesto
1000	AAAA800101	5000	2001-12-13	165	1000	Varilla 3/16	100	10
1000	AAAA800101	5000	2001-12-13	165	1010	Varilla 4/32	115	11.5
1000	AAAA800101	5000	2001-12-13	165	1020	Varilla 3/17	130	13
1000	AAAA800101	5000	2001-12-13	165	1030	Varilla 4/33	145	14.5

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

Se multiplican, enseñando todos los resultados de cada tabla.

### Construcción de consultas a partir de una especificación:

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

```
UPDATE entregan
SET fecha = DATE_FORMAT(fecha, '%d/%m/%Y')
WHERE fecha = DATE_FORMAT(fecha, '%Y/%m/%d')
Para cambiar el formato en MariaDB.
```

```
select descripcion from materiales,entregan
where materiales.clave = entregan.clave and fecha Between '01-01-2000' AND '31-12-2000'
```

¿Por qué aparecen varias veces algunas descripciones de material?

Porque un material puede ser entregado en diferentes días.

### Uso del calificador distinct:

¿Qué resultado obtienes en esta ocasión?

Lo mismo, pero sin repetirse.

### Ordenamiento:

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 denominacion,cantidad,fecha from proyectos, entregan
where proyectos.numero = entregan.numero
order by fecha DESC
```

### **Operadores de cadena:**

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

¿Qué resultado obtienes?

Todas las descripciones de materiales que la palabra comience por Si.

Explica que hace el símbolo '%'.

Empieza una cadena de caracteres.

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

Solo buscaría descripciones con Si

¿Qué resultado obtienes?

Ningún resultado.

Explica a qué se debe este comportamiento.

Porque en la tabla de materiales no existe ninguna descripción que solo sea Si.

¿Para qué sirve DECLARE?

Declarar variable Globales

¿Cuál es la función de @foo?

Guardar una cadena de caracteres

¿Que realiza el operador SET?

Asigna formato o valor dependiendo de que venga después del set.

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

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

Devuelve los RFCs de la tabla entregan que pertenezcan de A hasta la D

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

Devuelve los RFC que no posean la letra A

```
SELECT Numero FROM Entregan WHERE Numero LIKE ' 6' ;
```

Devuelve RFCs que terminen con el número 6

¿Cómo filtrarías rangos de fechas?

Con: between fecha1 and fecha2

```
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 hace la consulta?

Obtiene los RFCs de los proveedores con razón social que empiecen con la sílaba La

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

Para realizar una 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]
WHERE [numero] Between 5000 and 5010 AND RFC
IN (SELECT [rfc] FROM [Proveedores] WHERE razonsocial LIKE 'La%' and [entregan].[RFC] =
[proveedores].[RFC])
```

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

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

¿Qué hace?

Saca los dos elementos listados primero a tabla.

¿Qué sucede con la siguiente consulta?

```
SELECT TOP Numero FROM Proyectos
Error
```