**Consulta de una Tabla Completa**

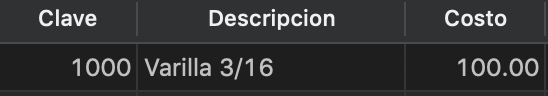
Select\* From Materiales



Muestra toda los datos que se encuentran en Materiales

**Selección**

Select\* From Materiales   
Where clave=1000

****

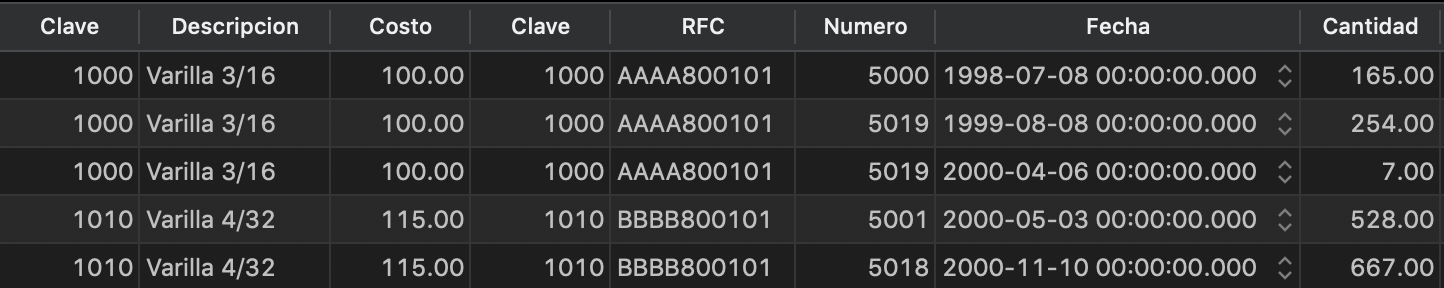
**Proyección**

Select clave,rfc,fecha from entregan

****

**Reunión Natural**

Select \* from Materiales,Entregan   
Where Materiales.clave = Entregan.clave

****

Si algún material no ha se ha entregado ¿Aparecería en el resultado de esta consulta? No ya que estamos haciendo una consulta de los materiales que ya fueron entregados.

**Reunión con criterio especifico**

Select \* From Entregan,Proyectos   
Where entregan.numero < = proyectos.numero

****

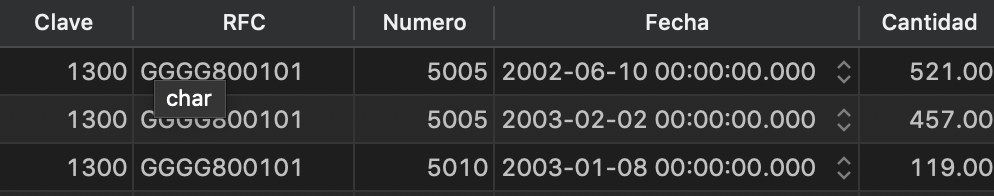
**Union**

(Select \* From Entregan Where Clave=1450)   
union   
(Select \* From Entregan Where Clave=1300)   
  


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

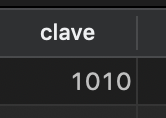
Select \* From Entregan

Where Clave =1450 OR Clave=1300



**Intersección**

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

****

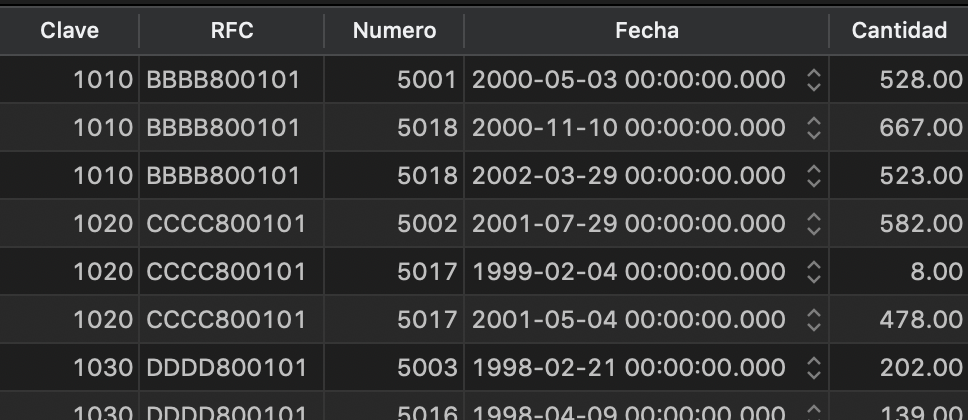
**Diferencia**

(Select \* From Entregan)   
minus   
(Select \* from Entregan where clave=1000)

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

SELECT\* FROM Entregan

Where clave <> 1000



**Plano Cartesiano**

Select \* From Entregan,Materiales

****

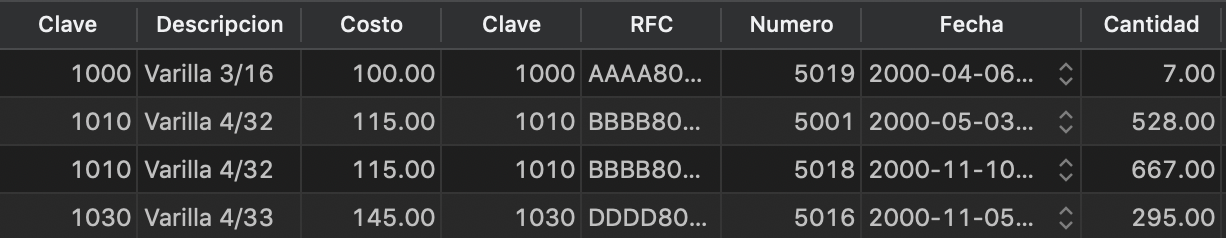
¿Cómo está definido el número de tuplas de este resultado en términos del número de tuplas de entregan y de materiales? Por cada tupla de entregan se hace una relacion por cada tupla de materiales.

**Construcción de consultas a partir de una especificación**

set dateformat dmy

Select Descripcion FROM Materiales,Entregan

Where materiales.clave = entregan.clave AND entregan.Fecha BETWEEN '01/01/00' AND '31/12/00'

****

¿Por qué aparecen varias veces algunas descripciones de material? Es porque el material fue entregado varias veces ese mismo año.

**Uso del calificador distinct**

Agrega la palabra distinct inmediatamente después de la palabra select a la consulta que planteaste antes.   
  
¿Qué resultado obtienes en esta ocasión? Que ya no se repiten los materiales, solo aparecen los materiales que fueron entregados en en ese año.

****

**Ordenamientos**

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 Entregan.numero,denominacion,fecha,cantidad FROM Proyectos,Entregan

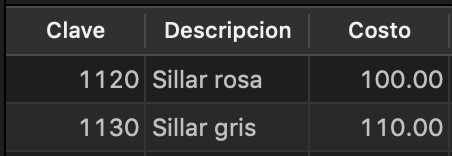
WHERE proyectos.numero = entregan.numero

ORDER BY fecha desc



**Operadores de cadena**

SELECT \* FROM productos where Descripcion LIKE 'Si%'

****

Explica que hace el símbolo '%'.  Que puede haber caracteres despues de lo insertado.

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

Traeria los registros que tienen una descripcion con nombre Si

¿Qué resultado obtienes?

Ningun registro.

Explica a qué se debe este comportamiento.

A que no hay ningun material que tenga la descripcion Si.

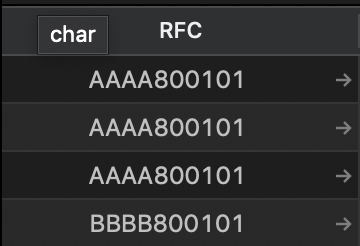
DECLARE @foo varchar(40);   
DECLARE @bar varchar(40);   
SET @foo = '¿Que resultado';   
SET @bar = ' ¿¿¿??? '   
SET @foo += ' obtienes?';   
PRINT @foo + @bar;   
  
**¿Qué resultado obtienes de ejecutar el siguiente código?**

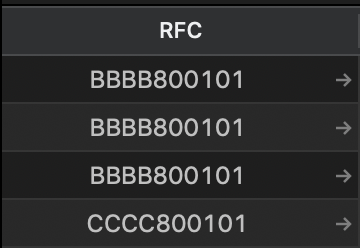
¿Que resultado obtienes? ¿¿¿??? **¿Para qué sirve DECLARE?**

Para declarar una variable **¿Cuál es la función de @foo?**

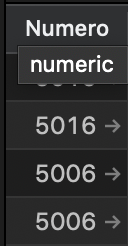
Poder concatenar dos resultados.  **¿Que realiza el operador SET?**

Establece el valor de una variable.

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


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


SELECT Numero FROM Entregan WHERE Numero LIKE '\_\_\_6';

****

**Operadores Lógicos**

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?

Selecciona el RFC Cantidad Fecha y Numero de la tabla Entregan en donde el numero este en los valores de 5000 y 5010 y que el RFC sea de los proveedores en donde la razon social empiece con las letras La.

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

Que los RFC que me regresan existan en en la tabla proveedores en donde la razon social sea igual a La…. Seguido por otras letras.

**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

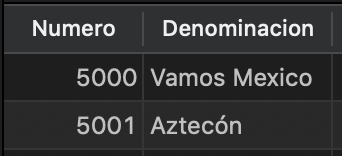
RFC IN ( SELECT [RFC]

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 TOP 2 \* FROM Proyectos



SELECT TOP Numero FROM Proyectos

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

Marcar un error ya que numero no es un entero al que le estemos diciendo que nos regrese los primeros registros de la tabla proyectos.

**Modificando la estructura de una tabla existente**

¿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 Entregan.Cantidad,Entregan.Clave,

(Entregan.Cantidad\*(Materiales.Costo))+(Entregan.Cantidad\*Materiales.Costo\*(Materiales.PorcentajeImpuesto/100) as ‘Total’

From Entregan, Materiales

Where Entregan.Clave = Materiales.Clave

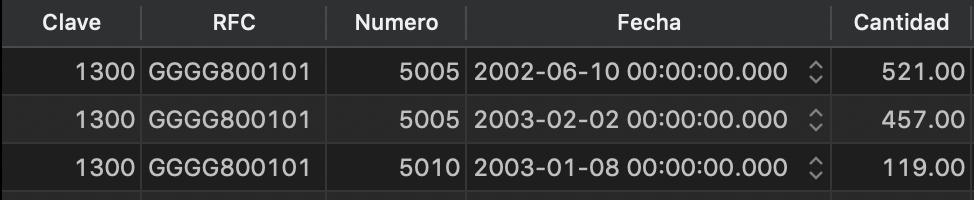
**Creación de Vistas**

CREATE VIEW claveUnion

AS

Select \* From Entregan

Where Clave =1450 OR Clave=1300



CREATE VIEW rfcCondicional

AS

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] )

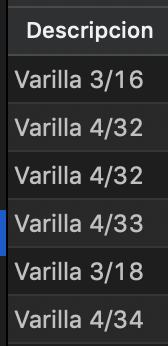


CREATE VIEW fechaMaterial

AS

Select Descripcion FROM Materiales,Entregan

Where materiales.clave = entregan.clave AND entregan.Fecha BETWEEN '01/01/00' AND '31/12/00'

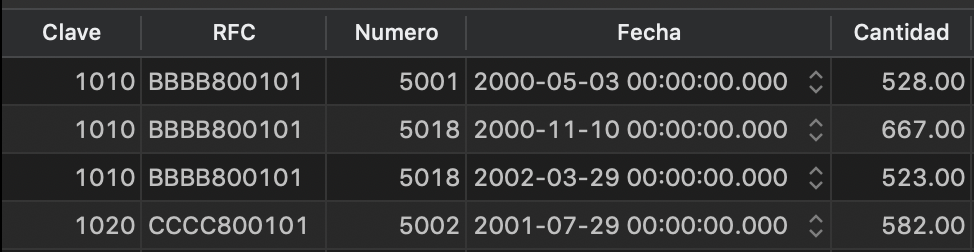


CREATE VIEW clavedif1000

AS

SELECT\* FROM Entregan

Where clave <> 1000

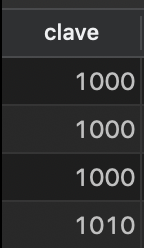


CREATE VIEW claveEntregan

AS

Select entregan.clave from Materiales,Entregan

Where Materiales.clave = Entregan.clave



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

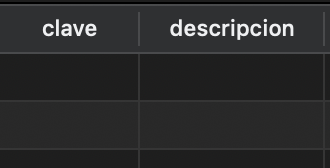
SELECT entregan.clave , descripcion FROM Materiales, Entregan, Proyectos

WHERE materiales.clave = entregan.clave AND entregan.numero = proyectos.numero AND proyectos.denominacion = 'Mexico sin ti no estamos completos'

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

SELECT Materiales.clave, descripcion FROM Proveedores,Entregan,Materiales

WHERE Proveedores.RFC= entregan.RFC AND materiales.clave = entregan.Clave AND proveedores.RazonSocial = 'Acme tools'



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

SELECT RFC FROM Entregan

WHERE Cantidad>=300 AND Fecha BETWEEN '01/01/00' AND '31/12/00'

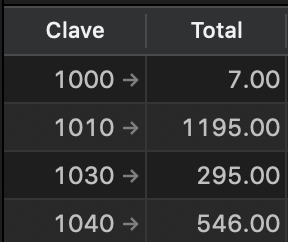


El Total entregado por cada material en el año 2000.

SELECT Clave, SUM(Cantidad) as 'Total'

FROM Entregan

WHERE Fecha Between '01-JAN-2000' AND '31-DEC-2000'

GROUP BY Clave  


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

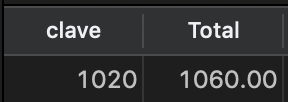
SELECT TOP 1 clave, SUM(Cantidad) as 'Total'

FROM Entregan

WHERE Fecha BETWEEN '01/01/01' AND '31/12/01'

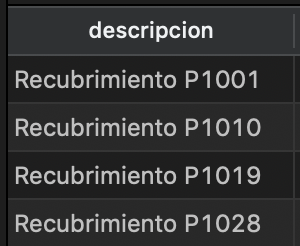
GROUP BY Clave

ORDER BY Total desc



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

SELECT descripcion FROM Materiales

WHERE descripcion LIKE '%ub%'  


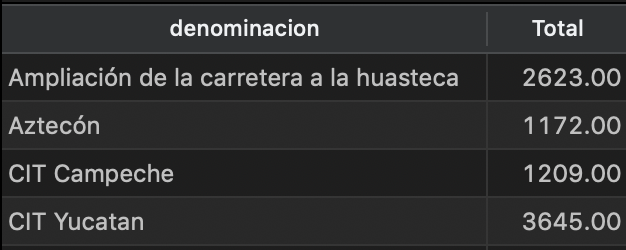
Denominación y suma del total a pagar para todos los proyectos.

SELECT denominacion, SUM(Cantidad) AS 'Total'

FROM Proyectos, Entregan, Materiales

WHERE proyectos.numero = entregan.Numero AND Materiales.Clave = Entregan.Clave

Group BY Denominacion

  
  
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 televisa AS

SELECT DISTINCT Denominacion, Entregan.RFC, RazonSocial

FROM Proyectos, Proveedores, Entregan

WHERE Entregan.RFC=Proveedores.RFC AND Proyectos.numero = Entregan.numero AND denominacion LIKE 'Televi%'

CREATE VIEW educandoCoahila AS

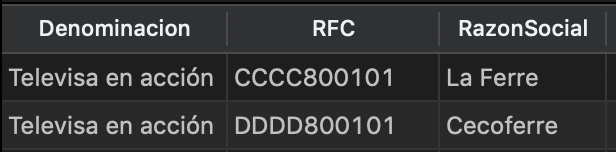
SELECT Entregan.RFC

FROM Proyectos, Proveedores, Entregan

WHERE Entregan.RFC=Proveedores.RFC AND Proyectos.numero = Entregan.numero AND denominacion LIKE 'Educ%'

SELECT \* FROM televisa

WHERE RFC NOT IN (SELECT RFC FROM educandoCoahila)



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 Denominacion, Entregan.RFC, RazonSocial

FROM Proyectos, Proveedores, Entregan

WHERE Entregan.RFC=Proveedores.RFC AND Proyectos.numero = Entregan.numero AND denominacion LIKE 'Televi%' AND Entregan.RFC

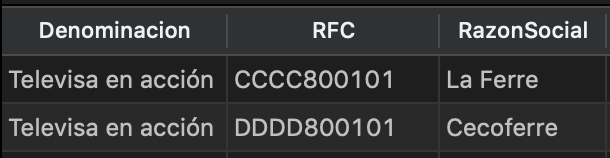
NOT IN(

SELECT Entregan.RFC

FROM Proyectos, Proveedores, Entregan

WHERE Entregan.RFC=Proveedores.RFC AND Proyectos.numero = Entregan.numero AND denominacion LIKE 'Educ%'

)



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 Materiales.Costo, Materiales.Descripcion

FROM Proyectos, Proveedores, Entregan

WHERE Entregan.RFC=Proveedores.RFC AND Proyectos.numero = Entregan.numero AND denominacion LIKE 'Televi%' AND Entregan.RFC

IN(

SELECT Entregan.RFC

FROM Proyectos, Proveedores, Entregan

WHERE Entregan.RFC=Proveedores.RFC AND Proyectos.numero = Entregan.numero AND denominacion LIKE 'Educ%'

)

|  |  |  |
| --- | --- | --- |
|  | Costo | Descripción |
| 1 | 50.0 | Ladrillos Rojos |
| 2 | 34.0 | Tepetate |

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

SELECT Materiales.Descripcion, SUM(Cantidad) as ‘Cantidad\_Entregados, SUM(Entregas.Cantidad\*(Materiales.Costo\*(PorcentajeImpuesto/100)+1)) as ‘CostoTotal’

FROM Entregan, Materiales, Proyectos

WHERE Materiales.Clave = Entregan.Clave AND Proyectos.Numero = Entregan.Numero

GROUP BY Materiales.Descripcion, Proyectos.Denominacion

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Descripcion | Denominacion | Cantidad\_Entregados | CostoTotal |
| 1 | Cantera rosa | Ampliación de la carretera a la huasteca | 162.00 | 5498.9280000 |
| 2 | Pintura C1010 | Ampliación de la carretera a la huasteca | 364.00 | 1601.600000 |

**II. Investigacion**

Imagen 1.1

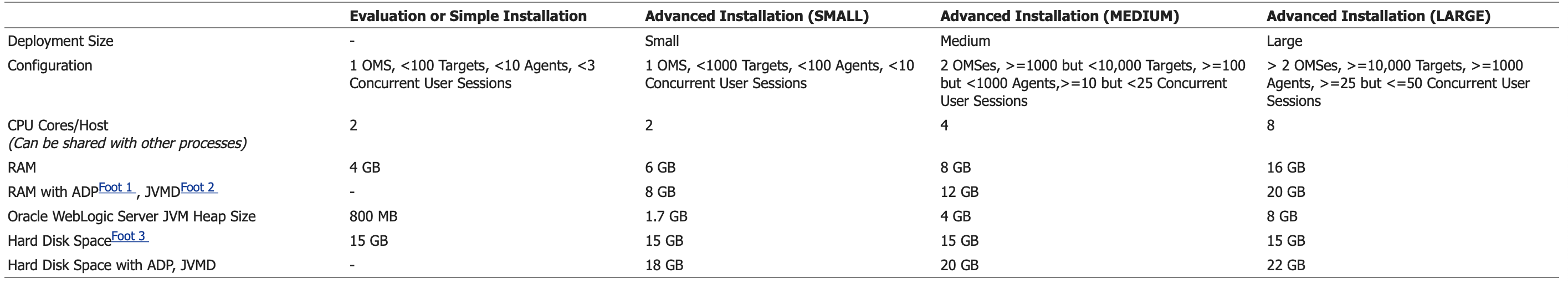
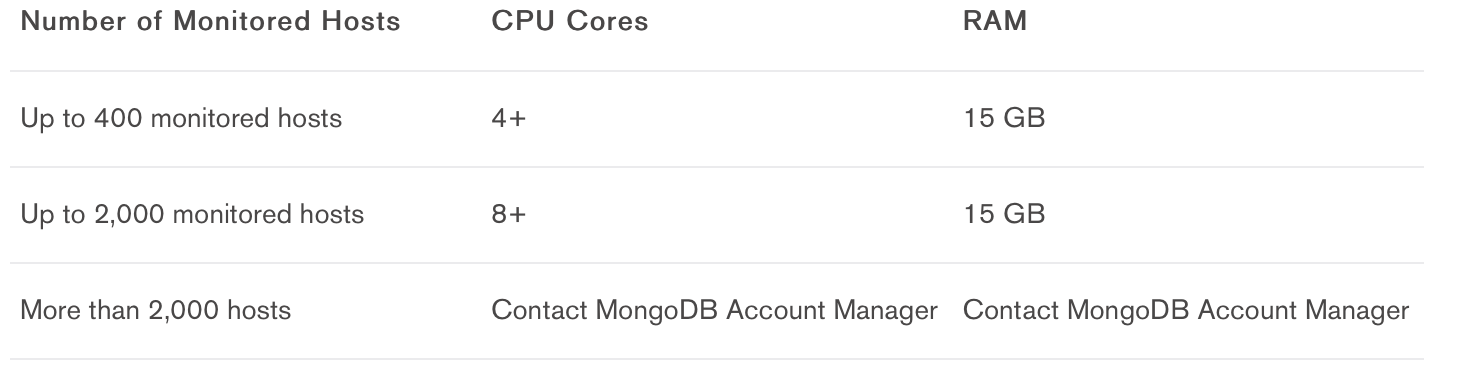
****

Imagen 2.1

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Oracle** | **SQL Server** | **MySql** | **Mongo DB** |
| Requerimientos Generales de Hardware | Imagen 1.1 | Processor - Minimum: 1 GHz (x86 processor) or 1.4 GHz (x64 processor) or 1.0 GHz (IA64 processor) Available Disk Space Minimum: 100 GB Free Disk Space on C: Drive  Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016. Or if dedicated PC acting as server: Windows 7 or 10 Professional (Minimum: 8 GB RAM) | CPU: Intel Core or Xeon 3GHz (or Dual Core 2GHz) or equal AMD CPU  Cores: Single (Dual/Quad Core is recommended)  RAM: 4 GB (6 GB recommended)  Graphic Accelerators: nVidia or ATI with support of OpenGL 1.5 or higher  Display Resolution: 1280×1024 is recommended, 1024×768 is minimum. | Imagen 2.1 |
| Ambientes o Plataformas en las que pueden operar | Windows, OSX, Linux, Solaris | Windows, Linux | Windows, Mac OSX, Ubuntu, Linux | Windows, Ubuntu, Amazon Linux |
| Costos de Implementación y Mantenimiento | Existe la version gratitua Oracle Express. Existen precios de memberias anuales que van desde los 10,000 hasta 20,000 dolares. | Existe una versión gratitua. Y tenemos la de pago que consta de 3,717 dolares y la enterprise que tiene un precio 14,256 dolares. | De 2,000 a 10,000 dolares. | Versión developer tiene precios que empiezan desde los 49 dolares. |
| Ventajas | Reconocido por su motor de base de datos y tiene la capacidad de trabajar con tamaños muy grandes de bases de datos. | Se puede administrar informacion de otros servidores de datos, cuenta con soporte de transacciones. | Su codigo es open source, es mur rapido al momento de realizar operaciones, gracias a los specs de seguridad que tienen hacen una buena opcion base de datos en internet. | Cuenta con motores de almacenamiento integrado y emplea estrcturas BSON. |
| Desventajas | Una de las grandes desventajas que se tienen es el precio que manejan en las licencias, no es tan fácil la comprensión del sistema. | Tiene muchas caracterisitcas que son limitadas, y el tamaño de memoria que utiliza de RAM es alto. | Porcentaje de las utiliades no estan documentadas y la compatibilidad con ACID es dependeinete del motor de almacenamiento. | Es tecnolgia muy joven, no es adecuada para transacciones complejas y no cuenta con un remplazo para soluciones de herencia. |
| Porcentaje del mercado que controlan | 3.67% | 13.25% | 33.08% | 4.64% |

https://www.oracle.com/assets/mysql-pricelist-183985.pdf

https://www.microsoft.com/es-mx/sql-server/sql-server-2017-pricing

<https://www.g2.com/products/mysql/pricing>

<https://www.mongodb.com/cloud/atlas/pricing?lang=es-es>

<https://docs.oracle.com/html/E22624_27/preinstall_req_hw.htm>

<https://www.miracleservice.com/product-features/system-requirements/>

<http://download.nust.na/pub6/mysql/doc/workbench/en/wb-requirements-hardware.html>

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