



Instituto Tecnológico de Estudios Superiores Monterrey

CAMPUS QUERÉTARO

Construcción de software y toma de decisiones

Ricardo Cortés Espinosa

Eduardo Daniel Juárez Pineda

Grupo 401

## **Laboratorio 20: Consultas en SQL**

**PRESENTA**

Alan Fernando Razo Peña - A01703350

Fecha:

8 de septiembre de 2022

## Consulta de un tabla completa

Álgebra relacional: Materiales

SQL: SELECT \* FROM Materiales

				clave	descripcion	precio	impuesto
<input type="checkbox"/>	Editar	Copiar	Borrar	1000	Varilla 3/16	100	10
<input type="checkbox"/>	Editar	Copiar	Borrar	1010	Varilla 4/32	115	11.5
<input type="checkbox"/>	Editar	Copiar	Borrar	1020	Varilla 3/17	130	13

## Selección

Álgebra relacional:  $SL\{clave=1000\}(materiales)$

SQL: select \* from materiales  
where clave=1000

clave	descripcion	precio	impuesto
1000	Varilla 3/16	100	10

## Proyección

Algebra relacional:  $PR\{clave, rfc, fecha\}(entregan)$

SQL: select clave, rfc, fecha from entregan

clave	rfc	fecha
1000	AAAA800101	2001-12-13
1200	EEEE800101	2003-03-15
1400	AAAA800101	1999-04-07
1010	BBBB800101	1998-07-28

## Reunión Natural

Algebra relacional: entregan  $\bowtie$  materiales

SQL: select \* from materiales, entregan  
where materiales.clave = entregan.clave

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

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

R= No, no va mostrar los no entregados porque solo pone los materiales que sí fueron entregados.

### Reunión con criterio específico

Algebra relacional:  $\text{entregan} \bowtie_{\text{entregan.numero} \leq \text{proyectos.numero}} \text{proyectos}$

SQL:

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

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	Aztecon
1200	EEEE800101	5000	2003-03-15	177	5001	Aztecon
1400	AAAA800101	5000	1999-04-07	382	5001	Aztecon
1010	BBB800101	5001	1998-07-28	528	5001	Aztecon
1210	FFF800101	5001	2000-05-21	43	5001	Aztecon

### Unión (se ilustra junto con selección)

Algebra relacional:  $\text{SL}\{\text{clave}=1450\}(\text{entregan}) \cup \text{SL}\{\text{clave}=1300\}(\text{entregan})$

SQL:

```
(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.

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

### Intersección (se ilustra junto con selección y proyección)

Algebra relacional.

$\text{PR}\{\text{clave}\}(\text{SL}\{\text{numero}=5001\}(\text{entregan})) \cap \text{PR}\{\text{clave}\}(\text{SL}\{\text{numero}=5018\}(\text{entregan}))$

SQL

Nota: Debido a que en SQL server no tiene definida alguna palabra reservada que nos permita hacer esto de una manera entendible, veremos esta sección en el siguiente laboratorio con el uso de Subconsultas. Un ejemplo de un DBMS que si tiene la

implementación de una palabra reservada para esta función es Oracle, en él si se podría generar la consulta con una sintaxis como la siguiente:

```
select clave from entregan where numero=5001 and clave
in
(select clave from entregan where numero=5018)
```

clave
1010

### Diferencia (se ilustra con selección )

Algebra relacional: entregan - SL{clave=1000}(entregan)

SQL:

```
select * from entregan
```

```
except
```

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

– Versión alterna –

```
select * from entregan
```

```
WHERE clave NOT IN
```

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

clave	rfc	numero	fecha	cantidad
1010	BBBB800101	5001	1998-07-28	528
1010	BBBB800101	5018	1997-02-09	523
1020	CCCC800101	5002	2003-12-16	582
1020	CCCC800101	5017	2000-03-29	8

### Producto cartesiano

Algebra relacional.

entregan X materiales

SQL

```
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?

R= Se multiplican las tuplas de entregan por las de 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

## Construcción de consultas a partir de una especificación

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

Recuerda que la fecha puede indicarse como '01-JAN-2000' o '01/01/00'.

```
SELECT descripcion
from entregan e, materiales m
WHERE fecha >= '2000 01-01'
AND fecha <= '2000-12-31'
```

### descripcion

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

Varilla 3/16

```
SELECT descripcion
from entregan e, materiales m
WHERE e.clave = m.clave
AND fecha >= '2000 01-01'
AND fecha <= '2000-12-31'
```

### descripcion

Recubrimiento P1028

Pintura B1021

Cantera blanca

Sillar gris

Pintura B1022

Block

Tubería 3.6

**Importante:** Recuerda que cuando vayas a trabajar con fechas, antes de que realices tus consultas debes ejecutar la instrucción "set dateformat dmy". Basta con que la ejecutes una sola vez para que el manejador sepa que vas a trabajar con ese formato de fechas.

¿Por qué aparecen varias veces algunas descripciones de material?

R= Porque se realizaron entregas del mismo material en diferentes fechas del año 2000.

### Uso del calificador distinct


En el resultado anterior, observamos que una misma descripción de material aparece varias veces.

Agrega la palabra distinct inmediatamente después de la palabra select a la consulta que planteaste antes.

```
SELECT DISTINCT descripcion  
from entregan e, materiales m  
WHERE fecha >= '2000 01-01'  
AND fecha <= '2000-12-31'
```

¿Qué resultado obtienes en esta ocasión?

R= Se enumeran los valores que son distintos, en este caso las descripciones y por lo tanto no se repiten.



descripcion
Varilla 3/16
Varilla 4/32
Varilla 3/17
Varilla 4/33
Varilla 3/18
Varilla 4/34
Varilla 3/19
Varilla 4/35

### Ordenamientos.

Si al final de una sentencia select se agrega la cláusula

order by campo [desc] [,campo [desc] ...]

donde las partes encerradas entre corchetes son opcionales (los corchetes no forman parte de la sintaxis), los puntos suspensivos indican que pueden incluirse varios campos y la palabra desc se refiere a descendente. Esta cláusula permite presentar los resultados en un orden específico.

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.

SQL:

```
SELECT p.numero, denominacion, fecha, cantidad
from proyectos p, entregan e
WHERE p.numero = e.numero
ORDER BY p.numero DESC, e.fecha DESC
```

numero	denominacion	fecha	cantidad
5019	Queretaro limpio	2004-11-19	94
5019	Queretaro limpio	2003-05-05	244
5019	Queretaro limpio	2000-11-10	107
5019	Queretaro limpio	1999-07-13	254
5019	Queretaro limpio	1998-05-27	199
5018	Tu cambio por la educación	2006-07-12	53
5018	Tu cambio por la educación	2006-04-15	302

## Uso de expresiones

En álgebra relacional los argumentos de una proyección deben ser columnas. Sin embargo en una sentencia SELECT es posible incluir expresiones aritméticas o funciones que usen como argumentos de las columnas de las tablas involucradas o bien constantes. Los operadores son:

+ Suma  
 - Resta  
 \* Producto  
 / División

Las columnas con expresiones pueden renombrarse escribiendo después de la expresión un alias que puede ser un nombre arbitrario; si el alias contiene caracteres que no sean números o letras (espacios, puntos etc.) debe encerrarse entre comillas dobles (" nuevo nombre" ). Para SQL Server también pueden utilizarse comillas simples.

## Operadores de cadena

El operador LIKE se aplica a datos de tipo cadena y se usa para buscar registros, es capaz de hallar coincidencias dentro de una cadena bajo un patrón dado.

También contamos con el operador comodín (%), que coincide con cualquier cadena que tenga cero o más caracteres. Este puede usarse tanto de prefijo como sufijo.

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

¿Qué resultado obtienes?

clave	descripcion	precio	impuesto
1110	Megablock	40	4

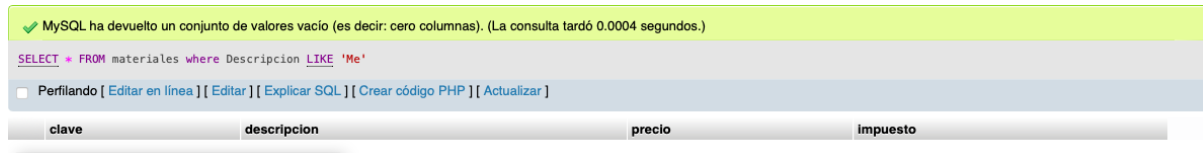
Explica que hace el símbolo '%'.

R= Es como un comodín que coincide con las letras que están antes o después de este.

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

R= No encontraría nada ya que no hay una descripción que coincide solo con 'Si'.

¿Qué resultado obtienes?



The screenshot shows a MySQL query interface. At the top, a green message bar states: "MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0004 segundos.)". Below this, the SQL query is displayed: `SELECT * FROM materiales where Descripción LIKE 'Me'`. Underneath the query, there are links for "Perfilando", "Editar en línea", "Editar", "Explicar SQL", "Crear código PHP", and "Actualizar". At the bottom, a table header is visible with columns: "clave", "descripcion", "precio", and "impuesto".

Explica a qué se debe este comportamiento.


A que el like hace referencia a lo que está contenido dentro de las comillas.

Otro operador de cadenas es el de concatenación, (+, +=) este operador concatena dos o más cadenas de caracteres.

Su sintaxis es : Expresión + Expresión.

Un ejemplo de su uso, puede ser: Un ejemplo de su uso, puede ser:

SELECT (precio + impuesto) as 'Precio final' FROM materiales



The screenshot shows a MySQL query result. The table has a single column labeled "Precio final". The data rows are: 110, 126.5, 143, 159.5, and 176.

```
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?**

R= ¿Que resultado obtienes¿¿¿???

**¿Para qué sirve DECLARE?**

R= Sirve para declarar la variable

**¿Cuál es la función de @foo?**

R= Marcador de posición de un valor que puede cambiar.

**¿Que realiza el operador SET?**

R= especifica las columnas que se deben actualizar y los valores nuevos para las columnas.



Sin embargo, tenemos otros operadores como [ ] , [^] y \_.

[ ] - Busca coincidencia dentro de un intervalo o conjunto dado. Estos caracteres se pueden utilizar para buscar coincidencias de patrones como sucede con LIKE.

[^] - En contra parte, este operador coincide con cualquier caracter que no se encuentre dentro del intervalo o del conjunto especificado.

\_ - El operador \_ o guion bajo, se utiliza para coincidir con un caracter de una comparación de cadenas.

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

```
SELECT RFC FROM Entregan WHERE RFC LIKE '[A-D]%' ; No me funcionó
```

Pero asi si:

```
SELECT RFC FROM Entregan WHERE RFC LIKE 'A%';
```

RFC
AAAA800101
AAAA800101
AAAA800101
AAAA800101

R= Lo que debería pasar aquí en la consulta original es que despliega en un rango de A a D todos los rfc's que inicien con dichas iniciales

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

R= Coincidir cualquier carácter que no está dentro del rango o especificación en los brackets.

```
SELECT Numero FROM Entregan WHERE Numero LIKE '____6';
```

R= regresa los números que terminan con 6

Numero
5006
5006
5006

### Operadores compuestos.

Los operadores compuestos ejecutan una operación y establecen un valor.

+ = (Suma igual)

- = (Restar igual)

\* = (Multiplicar igual)

/ = (Dividir igual)

% = (Módulo igual)

### **Operadores Lógicos.**

Los operadores lógicos comprueban la verdad de una condición, al igual que los operadores de comparación, devuelven un tipo de dato booleano (True, false o unknown).

**ALL** Es un operador que compara un valor numérico con un conjunto de valores representados por un subquery. La condición es verdadera cuando todo el conjunto cumple la condición.

**ANY o SOME** Es un operador que compara un valor numérico con un conjunto de valores. La condición es verdadera cuando al menos un dato del conjunto cumple la condición.

La sintaxis para ambos es: valor\_numerico {operador de comparación} subquery

**BETWEEN** Es un operador para especificar intervalos. Una aplicación muy común de dicho operador son intervalos de fechas.

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

¿Cómo filtrarías rangos de fechas?

Con: Between '2000-01-10' and '2000-12-31';

**EXISTS** Se utiliza para especificar dentro de una subconsulta la existencia de ciertas filas.

```
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?

Muestra las entregas de proveedores que empiecen con 'La'.

RFC	Cantidad	Fecha	Numero
AAAA800101	165	2001-12-13	5000
AAAA800101	382	1999-04-07	5000
CCCC800101	582	2003-12-16	5002
CCCC800101	603	2001-09-09	5002

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

Especificar la subconsulta que debería existir.

**IN** Especifica si un valor dado tiene coincidencias con algún valor de una subconsulta.

NOTA: Se utiliza dentro del WHERE pero debe contener un parametro. Ejemplo: Where proyecto.id IN Lista\_de\_Proyectos\_Subquery

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

Mostrando filas 0 - 11 (total de 12. La consulta tardó 0.0004 segundos.)

```
SELECT RFC,Cantidad, Fecha,Numero FROM Entregas WHERE Numero Between 5000 and 5010 AND RFC IN ( SELECT RFC FROM Proveedores WHERE Razonsocial LIKE 'L%' and Entregas.RFC = Proveedores.RFC )
```

Parámetro: [ Editar en línea ] [ Editar ] [ Explorar SQL ] [ Crear código PHP ] [ Actualizar ]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

Options	RFC	Cantidad	Fecha	Numero
<input type="checkbox"/> Editar	AAAA800101	165	2001-12-13	5000
<input type="checkbox"/> Editar	AAAA800101	382	1999-04-07	5000
<input type="checkbox"/> Editar	AAAA800101	582	2003-12-16	5002
<input type="checkbox"/> Editar	AAAA800101	603	2001-09-09	5002
<input type="checkbox"/> Editar	AAAA800101	115	2005-04-21	5010
<input type="checkbox"/> Editar	CCCC800101	582	2003-12-16	5002
<input type="checkbox"/> Editar	CCCC800101	468	2000-12-07	5009
<input type="checkbox"/> Editar	CCCC800101	699	2001-11-13	5010
<input type="checkbox"/> Editar	CCCC800101	480	2001-04-08	5006
<input type="checkbox"/> Editar	CCCC800101	624	2001-07-28	5009
<input type="checkbox"/> Editar	CCCC800101	603	2001-09-09	5002
<input type="checkbox"/> Editar	CCCC800101	278	1999-03-05	5008

**NOT** Simplemente niega la entrada de un valor booleano.

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.

Mostrando filas 0 - 24 (total de 25. La consulta tardó 0.0005 segundos.)

```
SELECT RFC,Cantidad, Fecha,Numero FROM Entregas WHERE Numero Between 5000 and 5010 AND RFC NOT IN ( SELECT RFC FROM Proveedores WHERE Razonsocial LIKE 'L%' and Entregas.RFC = Proveedores.RFC )
```

Parámetro: [ Editar en línea ] [ Editar ] [ Explorar SQL ] [ Crear código PHP ] [ Actualizar ]

1 | 25 | Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

Options	RFC	Cantidad	Fecha	Numero
<input type="checkbox"/> Editar	EEEE800101	177	2003-03-18	5000
<input type="checkbox"/> Editar	EEEE800101	328	1998-07-28	5001
<input type="checkbox"/> Editar	FFFF800101	43	2000-05-21	5001
<input type="checkbox"/> Editar	EEEE800101	981	2000-05-18	5001
<input type="checkbox"/> Editar	GGGG800101	24	2005-07-03	5002
<input type="checkbox"/> Editar	DDDD800101	265	1998-01-12	5003
<input type="checkbox"/> Editar	1888800101	530	1998-09-12	5003
<input type="checkbox"/> Editar	DDDD800101	576	2005-04-30	5003
<input type="checkbox"/> Editar	EEEE800101	263	1999-12-18	5004

El Operador **TOP**, es un operador que recorre la entrada, un query, y sólo devuelve el primer número o porcentaje específico de filas basado en un criterio de ordenación si es posible.

¿Qué hace la siguiente sentencia? Explica por qué.

```
SELECT * FROM Proyectos LIMIT 2
```

numero	denominacion
5000	Vamos Mexico
5001	Aztecon

R= Regresa las primeras dos tuplas de proyectos

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

```
SELECT Numero FROM Proyectos LIMIT 1
```

R= Regresa el primer número de las filas.

### Modificando la estructura de una tabla existente.

Agrega a la tabla materiales la columna PorcentajImpuesto con la instrucción:

```
ALTER TABLE materiales ADD PorcentajImpuesto NUMERIC(6,2);
```

A fin de que los materiales tengan un impuesto, les asignaremos impuestos ficticios basados en sus claves con la instrucción:

```
UPDATE materiales SET PorcentajImpuesto = 2*clave/1000;
```

esto es, a cada material se le asignará un impuesto igual al doble de su clave dividida entre diez.

Revisa la tabla de materiales para que compruebes lo que hicimos anteriormente.

clave	descripcion	precio	impuesto	PorcentajImpuesto
1000	Varilla 3/16	100	10	2.00
1010	Varilla 4/32	115	11.5	2.02
1020	Varilla 3/17	130	13	2.04

¿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?

Mostrando filas 0 - 19 (total de 20, la consulta tardó 0.0004 segundos)

ejemplo: numero, sub-cantidad \* precio = cantidad \* precio \* Porcentaje[etiqueta]: as ImporteTotal from materiales, entregas where materiales.clave = entregas.clave group by entregas.numero;

Partiendo [Editar en línea] [Editar] [Explor SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

numero	ImporteTotal
SC00	975362
SC01	327716.9
SC02	942963.2
SC03	917687.4
SC04	2044227.4
SC05	1274061.6
SC06	729092.4
SC07	476131.36
SC08	358012.08
SC09	570268.2
SC10	500875.9
SC11	795249.7
SC12	3058932.7
SC13	1625941.5
SC14	862796.6
SC15	2823930
SC16	1848685.9
SC17	488148.8
SC18	1664872.1
SC19	1005453.1

## Creación de vistas

La sentencia:

Create view nombrevista (nombrecolumna1 , nombrecolumna2 ,..., nombrecolumna3 ) as select...

Permite definir una vista. Una vista puede pensarse como una consulta etiquetada con un nombre, ya que en realidad al referirnos a una vista el DBMS realmente ejecuta la consulta asociada a ella, pero por la cerradura del álgebra relacional, una consulta puede ser vista como una nueva relación o tabla, por lo que es perfectamente válido emitir la sentencia:

select \* from nombrevista

¡Como si nombrevista fuera una tabla!

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.

```
1 Create view Vistal (RFC , RazonSocial)
2 as select rfc, RazonSocial from proveedores;
3 select * from Vistal;
```

✓ Mostrando filas 0 - 7 (total de 8, La consulta tardó 0.0005 segundos.)

```
select * from Vistal;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

				RFC	RazonSocial
<input type="checkbox"/>				AAAA800101	La fragua
<input type="checkbox"/>				BBBB800101	Oviedo
<input type="checkbox"/>				CCCC800101	La Ferre
<input type="checkbox"/>				DDDD800101	Cecoferre
<input type="checkbox"/>				EEEE800101	Alvin
<input type="checkbox"/>				FFFF800101	Comex
<input type="checkbox"/>				GGGG800101	Tabiguera del centro
<input type="checkbox"/>				HHHH800101	Tubasa

La parte (nombrecolumna1,nombrecolumna2,.de la sentencia create view puede ser omitida si no hay ambigüedad en los nombres de las columnas de la sentencia select asociada.

Importante: Las vistas no pueden incluir la cláusula order by.

A continuación se te dan muchos enunciados de los cuales deberás generar su correspondiente consulta.

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".

✓ Mostrando filas 0 - 2 (total de 3, La consulta tardó 0.0004 segundos.)

```
select clave, descripcion from materiales where materiales.clave in (select clave from entregan, proyectos where entregan.numero = proyectos.numero and proyectos.denominacion = 'México sin ti no estamos completos');
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:  | Sort by key: Ninguna

+ Opciones

				clave	descripcion
<input type="checkbox"/>				1030	Varilla 4/33
<input type="checkbox"/>				1230	Cemento
<input type="checkbox"/>				1430	Pintura B1022

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

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0.0004 segundos.)

```
select clave, descripcion from materiales where materiales.clave in (select clave from entregas, proveedores where entregas.rfc = proveedores.rfc and proveedores.razonsocial = 'Acme tools');
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

clave | descripcion

Operaciones sobre los resultados de la consulta

[Crear vista](#)

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

✓ Mostrando filas 0 - 2 (total de 3, La consulta tardó 0.0004 segundos.)

```
select proveedores.rfc, avg(cantidad) from proveedores, entregas where proveedores.rfc = entregas.rfc and year(entregas.fecha) = 2000 group by entregas.rfc having avg(cantidad) >= 300;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

	rfc	avg(cantidad)
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	BBBBB00101	601.0000
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	FFFFF00101	322.7500
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	GGGGG00101	583.0000

El Total entregado por cada material en el año 2000.

✓ Mostrando filas 0 - 9 (total de 10, La consulta tardó 0.0003 segundos.)

```
select descripcion, sum(cantidad) as Total from materiales, entregas where materiales.clave = entregas.clave and year(entregas.fecha) = 2000 group by descripcion;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

descripcion	Total
Block	466
Cantera blanca	583
Pintura B1021	708
Pintura B1022	13
Pintura C1010	265
Recubrimiento P1028	43
Sillar gris	625
Tubería 3.6	72
Varilla 3/17	8
Varilla 4/34	623

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

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0.0011 segundos.)

```
SELECT clave, MAX(ventas) from ventas2001;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

clave	MAX(ventas)
1100	1091

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

✓ Mostrando filas 0 - 11 (total de 12, La consulta tardó 0.0002 segundos.)

```
select descripcion from materiales where descripcion like '%ub%';
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

	descripcion
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Recubrimiento P1001
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Recubrimiento P1010
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Recubrimiento P1019
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Recubrimiento P1028
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Recubrimiento P1037
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 3.5
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 4.3
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 3.6
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 4.4
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 3.7
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 4.5
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Tubería 3.8

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

✓ Mostrando filas 0 - 19 (total de 20, La consulta tardó 0.0006 segundos.)

```
select proyectos.denominacion, sum(materiales.precio * entregan.cantidad + materiales.precio * entregan.cantidad * materiales.PorcentajeImpuesto) as Total from materiales, entregan, proyectos where materiales.clave = entregan.clave and entregan.numero = proyectos.numero group by proyectos.denominacion;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

	denominacion	Total
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Ampliación de la carretera a la huasteca	1948685.9
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Aztecon	507116.9
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	CIT Campeche	542563.2
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	CIT Yucatan	2833930
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Construcción de Hospital Infantil	500875.3
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Construcción de plaza Magnolias	415131.34
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Disco Atlantic	570266.2
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Educando en Coahuila	2044227.4
<input type="checkbox"/> <a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	Intervent Durango	1074051.5

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).



Mostrando filas 0 - 2 (total de 3, La consulta tardó 0.0005 segundos.)

```
select rfc,razonsocial, denominacion from proveedores where proveedores.rfc not in (select rfc from proveedoreseducando);
```

☐ Perfilando [\[ Editar en línea \]](#) [\[ Editar \]](#) [\[ Explicar SQL \]](#) [\[ Crear código PHP \]](#) [\[ Actualizar \]](#)

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

rfc	razonsocial	denominacion
DDDD800101	Cecofeme	Televisa en acción
DDDD800101	Cecofeme	Televisa en acción
CCCC800101	La Ferre	Televisa en acción

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).

Mostrando filas 0 - 2 (total de 3, La consulta tardó 0.0005 segundos.)

```
select proveedores.rfc, razonsocial, denominacion from proveedores, proyectos, entregan where proveedores.rfc = entregan.rfc and proyectos.numero = entregan.numero and denominacion = 'Televisa en accion' and proveedores.rfc not in (select proveedores.rfc from proveedores, proyectos, entregan where proveedores.rfc = entregan.rfc and proyectos.numero = entregan.numero and denominacion = 'Educando en Coahuila');
```

☐ Perfilando [\[ Editar en línea \]](#) [\[ Editar \]](#) [\[ Explicar SQL \]](#) [\[ Crear código PHP \]](#) [\[ Actualizar \]](#)

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

rfc	razonsocial	denominacion
DDDD800101	Cecofeme	Televisa en acción
DDDD800101	Cecofeme	Televisa en acción
CCCC800101	La Ferre	Televisa en acción

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.

Mostrando filas 0 - 24 (total de 90, La consulta tardó 0.0009 segundos.)

```
select descripcion, precio from materiales, proveedores, proyectos, entregan where proveedores.rfc = entregan.rfc and proyectos.numero = entregan.numero and denominacion = 'Televisa en accion' and proveedores.rfc in (select proveedores.rfc from proveedores, proyectos, entregan where proveedores.rfc = entregan.rfc and proyectos.numero = entregan.numero and denominacion = 'Educando en Coahuila');
```

☐ Perfilando [\[ Editar en línea \]](#) [\[ Editar \]](#) [\[ Explicar SQL \]](#) [\[ Crear código PHP \]](#) [\[ Actualizar \]](#)

1 > >> | ☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

descripcion	precio
Vanilla 3/16	100
Vanilla 3/16	100
Vanilla 4/32	115
Vanilla 4/32	115
Vanilla 3/17	130
Vanilla 3/17	130
Vanilla 3/17	130
Vanilla 3/17	130