



Instituto Tecnológico de Estudios Superiores Monterrey

CAMPUS QUERÉTARO

Laboratorio 20

AUTOR

Jose Armando Rosas Balderas

A01704132

Consulta de un tabla completa

Algebra relacional

materiales

SQL

select * from materiales

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The left sidebar lists databases like 'New', 'entregas', 'materiales', 'proveedores', 'proyectos', and others. The 'materiales' table is selected in the main area. The SQL tab contains the query: 'SELECT * FROM materiales;'. The results show 45 rows of data with columns: clave, descripcion, precio, and impuesto. The data is as follows:

	clave	descripcion	precio	impuesto
1	1000	Varilla 3/16	100	10
2	1010	Varilla 4/32	115	11.5
3	1020	Varilla 3/17	130	13
4	1030	Varilla 4/33	145	14.5
5	1040	Varilla 3/18	160	16
6	1050	Varilla 4/34	175	17.5
7	1060	Varilla 3/19	190	19
8	1070	Varilla 4/35	205	20.5

Selección

Algebra relacional

SL{clave=1000}(materiales)

SQL

select * from materiales

where clave=1000

The screenshot shows the phpMyAdmin interface for a database named 'entregas'. The 'materiales' table is selected. The query entered is 'select * from materiales where clave=1000;'. The results show one row with the following data:

clave	descripcion	precio	impuesto
1000	Varilla 3/16	100	10

Below the table, there are buttons for Edit, Copy, Delete, and Export.

Proyección

Algebra relacional

PR{clave,rfc,fecha} (entregan)

SQL

select clave,rfc,fecha from entregan

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The left sidebar lists databases like 'information_schema', 'mysql', and 'test', and tables such as 'entregas', 'materiales', 'proveedores', and 'proyectos'. The main area displays the results of a SELECT query:

```
USE entregas;
select clave, rfc, fecha from entregan;
```

The results table shows the following data:

	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
<input type="checkbox"/>	1030	DDDD800101	1998-01-12

Reunión Natural

Algebra relacional

entregan JN materiales

SQL

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

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The current table is 'materiales'. The left sidebar shows the database structure with tables like 'entregas', 'materiales', and 'proveedores'. The main area displays the results of the executed SQL query:

```
USE entregas;  
select * from materiales,entregan where materiales.clave = entregan.clave;
```

The results show 24 rows of data:

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
1030	Varilla 4/33	145	14.5	1030	DDDD800101	5003	1998-01-12	202

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

- No

Reunión con criterio específico

Algebra relacional

entregan JN{entregan.numero <= proyectos.numero} proyectos

SQL

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

The screenshot shows the phpMyAdmin interface for a database named 'entregas'. The left sidebar lists various databases and tables. The main area displays the results of a SQL query:

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

The results table shows the following data:

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	BBBB800101	5001	1998-07-28	528	5001	Aztecon

Unión (se ilustra junto con selección)

Algebra relacional

$\text{SL}\{\text{clave}=1450\}(\text{entregan}) \text{ UN } \text{SL}\{\text{clave}=1300\}(\text{entregan})$

SQL

(select * from entregan where clave=1450)

union

(select * from entregan where clave=1300)

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The 'Table' tab is selected, showing the 'entregan' table. The SQL query entered is:

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

The results pane displays two rows of data from the 'entregan' table:

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?

- select * from entregan where clave=1450 OR clave = 1300;

Intersección (se ilustra junto con selección y proyección)

Algebra relacional

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

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

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** entregas
- Table:** entregan
- Query Results:**
 - MySQL returned an empty result set (i.e. zero rows). (Query took 0.0002 seconds.)
 - USE entregas;
 - [Edit inline] [Edit] [Create PHP code]
⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
 - Showing rows 0 - 0 (1 total). Query took 0.0005 seconds.
 - (`select clave from entregan where numero=5001`) intersect (`select clave from entregan where numero=5018`);
- Table Structure:** Shows the 'clave' column with value 1010.

Diferencia (se ilustra con selección)

Algebra relacional

entregan - SL{clave=1000}(entregan)

SQL

(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 NOT IN (1000)
- SELECT * FROM entregan
 WHERE clave NOT IN (SELECT clave FROM entregan WHERE clave= 1000);

phpMyAdmin

Server: 127.0.0.1 » Database: entregas » Table: entregar

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show query box

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0002 seconds.)

USE entregas;

[Edit inline] [Edit] [Create PHP code]

Showing rows 0 - 24 (85 total, Query took 0.0004 seconds.)

SELECT * FROM entregar WHERE clave NOT IN (1000);

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	clave	rfc	numero	fecha	cantidad
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1010	BBBB800101	5001	1998-07-28	528
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1010	BBBB800101	5018	1997-02-09	523
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1020	CCCC800101	5002	2003-12-16	582
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1020	CCCC800101	5017	2000-03-29	8
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1030	DDDD800101	5003	1998-01-12	202
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1030	DDDD800101	5016	2005-06-07	295
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1040	EEEE800101	5004	1999-12-18	263
<input checked="" type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1040	EEEE800101	5015	1999-05-29	540

Console Edit Copy Delete

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?

El numero de tuplas de la tabla entregan multiplicado por el numero de tuplas de la tabla materiales.

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The current table is 'entregan'. The results of the query 'select * from entregan,materiales;' are displayed in a grid. The grid has 16 columns (8 from 'entregan' and 8 from 'materiales') and 3914 rows. The columns are labeled: clave, rfc, numero, fecha, cantidad, clave, descripcion, precio, and impuesto. The data includes various product codes like 1000 AAAA800101 and 1000 AAAB800101, along with their descriptions such as 'Vanilla 3/16', 'Vanilla 4/32', etc., and their prices and taxes.

clave	rfc	numero	fecha	cantidad	clave	descripcion	precio	impuesto
1000	AAAA800101	5000	2001-12-13	165	1000	Vanilla 3/16	100	10
1000	AAAAB800101	5000	2001-12-13	165	1010	Vanilla 4/32	115	11.5
1000	AAAA800101	5000	2001-12-13	165	1020	Vanilla 3/17	130	13
1000	AAAA800101	5000	2001-12-13	165	1030	Vanilla 4/33	145	14.5
1000	AAAAB800101	5000	2001-12-13	165	1040	Vanilla 3/18	160	16
1000	AAAAB800101	5000	2001-12-13	165	1050	Vanilla 4/34	175	17.5
1000	AAAAB800101	5000	2001-12-13	165	1060	Vanilla 3/19	190	19
1000	AAAAB800101	5000	2001-12-13	165	1070	Varilla 4/35	205	20.5
1000	AAAAB800101	5000	2001-12-13	165	1080	Ladrillos rojos	50	5
1000	AAAAB800101	5000	2001-12-13	165	1090	Ladrillos grises	35	3.5
1000	AAAAB800101	5000	2001-12-13	165	1100	Block	30	3
1000	AAAAB800101	5000	2001-12-13	165	1110	Megablock	40	4
1000	AAAAB800101	5000	2001-12-13	165	1120	Sillar rosa	100	10
1000	AAAAB800101	5000	2001-12-13	165	1130	Sillar gris	110	11

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'.

Query

```
SELECT m.descripcion, FROM entregan e, materiales m  
WHERE e.clave = m.clave  
AND e.fecha BETWEEN '2000/01/01' AND '2000/12/31';
```

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** entregas
- Table:** materiales
- Query:** SELECT m.descripcion FROM entregan e, materiales m WHERE e.clave = m.clave AND e.fecha BETWEEN '2000/01/01' AND '2000/12/31';
- Results:** 25 rows displayed, with a total of 34 rows found.
- Table Structure:** The 'materiales' table has columns: descripcion (text), cantidad (int), clave (pri, int), fecha (pri, date), numero (pri, int), and rfc (pri, varchar).
- Rows:** The results list various material descriptions, such as 'Varilla 3/17', 'Varilla 4/34', 'Block', 'Sillar gris', 'Cantera blanca', 'Recubrimiento P1028', 'Tuberia 3.6', 'Pintura C1010', 'Pintura B1021', and 'Pintura B1022'.

¿Por qué aparecen varias veces algunas descripciones de material?

- Porque el mismo material se entregó en diferentes fechas

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.

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The current table is 'materiales'. The SQL query executed is:

```
SELECT DISTINCT m.descripcion FROM entregan e, materiales m WHERE e.clave = m.clave AND e.fecha BETWEEN '2000/01/01' AND '2000/12/31';
```

The results show 9 unique descriptions from the 'materiales' table:

descripcion
Vanilla 3/17
Vanilla 4/34
Block
Siller gris
Cantera blanca
Recubrimiento P1028
Tubería 3 6
Pintura C1010
Pintura B1021
Pintura B1022

¿Qué resultado obtienes en esta ocasión?

La descripción de los materiales aparecen solo una vez.

Ordenamientos

Si al final de una sentencia select se agrega la cláusula

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

dónde 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.

Query

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

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** entregas
- Table:** entregan
- Query Result:**
 - Showing rows 0 - 24 (87 total, Query took 0.0007 seconds.) [numero: 5000... - 5007...] [fecha: 2003-03-15... - 2005-06-06...]
 - SQL Query:
`SELECT p.numero, p.denominacion, e.fecha, e.cantidad FROM proyectos p, entregan e WHERE p.numero = e.numero ORDER BY p.numero, e.fecha DESC;`
 - Buttons: Profiling, Edit inline, Explain SQL, Create PHP code, Refresh
- Table Data:** A grid showing the results of the query. The columns are numero, denominacion, fecha, and cantidad. The data includes various project numbers (e.g., 5000, 5001, 5002) with their corresponding denominations (e.g., Vamos Mexico, Aztecon, CIT Campeche), dates (e.g., 2003-03-15, 2005-07-03), and quantities (e.g., 177, 24).

Uso de expresiones.

- + Suma
- Resta
- * Producto
- / División

Operadores de cadena

Query

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

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The left sidebar shows the database structure with tables like 'entregas', 'materiales', and 'proveedores'. The main area displays the results of a query:

```
Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)  
SELECT * FROM materiales where Descripcion LIKE 'Si%';
```

The results table shows two rows:

	clave	descripcion	precio	impuesto
	1120	Sillar rosa	100	10
	1130	Sillar gris	110	11

At the bottom, there are options to bookmark the query or copy it to the clipboard.

¿Qué resultado obtienes?

- Todos los materiales cuya descripción comience con 'Si'

Explica qué hace el símbolo '%'.

- 0 o más caracteres

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

- Despliega todos los materiales cuya descripción sea 'Si'

¿Qué resultado obtienes?

- No obtengo ningún dato

Explica a qué se debe este comportamiento.

- No hay material cuya descripción sea simplemente 'Si'

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

¿Qué resultado obtienes de ejecutar el siguiente código?

- No está implementado
- ¿Qué resultado obtienes? ¿¿¿???

¿Para qué sirve DECLARE?

- Declaración de una variable local

¿Cuál es la función de @foo?

- Variable de tipo VARCHAR(40)

¿Que realiza el operador SET?

- Asigna un valor a una variable

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

SELECT RFC FROM Entregan WHERE RFC LIKE '[AD]%';

- Selecciona todos RFC que comiencen con A, B, C o D.

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

- Selecciona todos RFC que no comiencen con A.

SELECT Numero FROM Entregan WHERE Numero LIKE '___6';

- Selecciona todos los numeros de entregan cuyos numeros de 4 cifras que terminen en 6

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The current table is 'Entregan'. The query executed was 'SELECT Numero FROM Entregan WHERE Numero LIKE '___6';'. The results show nine rows of data, all with the value '5016' in the 'Numero' column. Each row has edit, copy, and delete options.

	Numero
<input type="checkbox"/>	5016
<input type="checkbox"/>	5006
<input type="checkbox"/>	5006
<input type="checkbox"/>	5016
<input type="checkbox"/>	5016
<input type="checkbox"/>	5006
<input type="checkbox"/>	5006
<input type="checkbox"/>	5016
<input type="checkbox"/>	5016

Operadores compuestos.

+ = (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 el operador BETWEEN

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?

- De todos los registros seleccionados en la proyección, solamente mostrara los registros de “entregan” cuya razon social de los proveedores comience con “La”.

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

- Delimitar la subconsulta.

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

Query

```
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);
```

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.

Query

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

Realiza un ejemplo donde apliques algún operador : ALL, SOME o ANY.

Query

```
SELECT RFC,Cantidad, Fecha, Numero
```

```
FROM Entregan
```

```
WHERE Numero > ANY (SELECT Numero FROM Entregan WHERE Numero = 5000)
```

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.

Ejecuta la consulta y de los registros encontrados solo devuelve 2.

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

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

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

- La consulta no se ejecuta debido a errores de sintaxis, ya que se escribe la palabra reservada para establecer un límite pero no se especifica el número.

```
SELECT TOP Numero FROM Proyectos;
```

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

Modificando la estructura de un tabla existente.

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

ALTER TABLE materiales ADD PorcentajeImpuesto 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 PorcentajeImpuesto = 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.

¿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?

Query

```
SELECT ((e.cantidad*m.precio)*((100 - m.PorcentajeImpuesto)/100)) AS 'Importe entregas'  
FROM materiales m, entregan e  
WHERE m.clave = e.clave
```

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.

Query 1

```
CREATE VIEW importeVentas AS SELECT ((e.cantidad*m.precio)*((100 -  
m.PorcentajeImpuesto)/100)) AS 'Importe entregas'
```

```
FROM materiales m, entregan e
```

```
WHERE m.clave = e.clave
```

```
SELECT * FROM ImporteVentas
```

Query 2

```
CREATE VIEW razonSocialLaV1 AS 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);
```

```
SELECT * FROM razonSocialLaV1;
```

Query 3

```
CREATE VIEW razonSocialLaV2 AS SELECT RFC,Cantidad, Fecha,Numero  
FROM Entregan
```

```
WHERE Numero Between 5000 and 5010 AND
```

```
RFC NOT IN (SELECT RFC
```

```
FROM Proveedores
```

```
WHERE RazonSocial NOT LIKE 'La%' and Entregan.RFC = Proveedores.RFC);
```

```
SELECT * FROM razonSocialLaV2;
```

Query 4

```
CREATE VIEW threeDigitsAndSix AS SELECT Numero FROM Entregan WHERE Numero  
LIKE '___6';
```

```
SELECT * FROM threeDigitsAndSix;
```

Query 5

```
CREATE VIEW sortedProjects AS SELECT p.numero, p.denominacion, e.fecha, e.cantidad  
FROM proyectos p, entregan e  
WHERE p.numero = e.numero  
ORDER BY p.numero, e.fecha DESC;
```

```
SELECT * FROM sortedProjects;
```

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".

Query

```
SELECT m.clave, m.descripcion  
FROM materiales m, entregan e, proyectos p  
WHERE m.clave = e.clave  
AND e.numero = p.numero  
AND p.denominacion = 'México sin ti no estamos completos';
```

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** entregas
- Table:** materiales
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Message Bar:** Showing rows 0 - 2 (3 total, Query took 0.0007 seconds.)
- Query Box:** SELECT m.clave, m.descripcion FROM materiales m, entregan e, proyectos p WHERE m.clave = e.clave AND e.numero = p.numero AND p.denominacion = 'México sin ti no estamos completos'.
- Buttons:** Profiling, Edit inline, Explain SQL, Create PHP code, Refresh.
- Table View:** Shows the results of the query with columns clave and descripcion. The data is:

	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".

Query

```
SELECT m.clave, m.descripcion  
FROM materiales m, entregan e, proveedores p  
WHERE m.clave = e.clave  
AND e.rfc = p.rfc  
AND p.razonsocial = 'Acme tools';
```

The screenshot shows the MySQL Workbench interface. The top navigation bar includes tabs for 'Server: 127.0.0.1', 'Database: entregas', and 'Table: materiales'. Below the navigation bar is a toolbar with various database management functions like 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers'. A sub-toolbar below the main toolbar contains buttons for 'SELECT *', 'SELECT', 'INSERT', 'UPDATE', 'DELETE', 'Clear', 'Format', and 'Get auto-saved query'. There are also checkboxes for 'Bind parameters' and 'Bookmark this SQL query'. The main area displays a message: 'MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)' followed by the SQL query: /* Los materiales (clave y descripción) que han sido proporcionados por el proveedor "Acme tools".*/ SELECT m.clave, m.descripcion FROM materiales m, entregan e, proveedores p WHERE e.rfc = p.rfc AND p.rfc = 'Acme tools'; Below the message are buttons for '[Edit inline]', '[Edit]', and '[Create PHP code]'. A tab labeled 'clave' is selected. A 'Query results operations' section contains a 'Create view' button. Further down are buttons for 'Bookmark this SQL query' (with a 'Label:' input field), 'Let every user access this bookmark', and another 'Bookmark this SQL query' button.

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

Query

```
SELECT p.rfc
FROM entregan e, proveedores p
WHERE e.rfc = p.rfc
AND e.fecha BETWEEN '2000/01/01' AND '2000/12/31'
GROUP BY p.rfc
HAVING AVG(e.cantidad) > 300;
```

Server: 127.0.0.1 » Database: entregas » Table: proveedores

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Delimiter : Show this query here again Retain query box Rollback when finished Enable foreign key checks Go Hide query box

Showing rows 0 - 2 (3 total, Query took 0.0014 seconds)

```
/* El RFC de los proveedores que durante el 2000 entregaron en promedio cuando menos 300 materiales.*/
SELECT p.rfc FROM entregan e, proveedores p WHERE e.fecha BETWEEN '2000/01/01' AND '2000/12/31' GROUP BY p.rfc HAVING AVG(e.cantidad) > 300;
```

[Edit inline] [Edit] [Create PHP code]

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

Extra options

rfc
BBBB800101
FFFF800101
GGGG800101

Check all With selected:

El Total entregado por cada material en el año 2000.

Query

```
SELECT m.clave, SUM(e.cantidad) AS 'Total'
FROM entregan e, materiales m
WHERE e.clave = m.clave
AND e.fecha BETWEEN '2000/01/01' AND '2000/12/31'
GROUP BY m.clave;
```

Server: 127.0.0.1 » Database: entregas » Table: materiales

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers Hide query box

Showing rows 0 - 10 (11 total, Query took 0.0005 seconds.)

```
/* El Total entregado por cada material en el año 2000. */
SELECT m.clave, SUM(e.cantidad) AS 'Total'
FROM entregan e, materiales m
WHERE e.clave = m.clave AND e.fecha BETWEEN '2000/12/31' GROUP BY m.clave;
```

[Edit inline] [Edit] [Create PHP code]

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

Extra options

	clave	Total
<input type="checkbox"/>	1020	8
<input type="checkbox"/>	1050	623
<input type="checkbox"/>	1100	466
<input type="checkbox"/>	1130	625
<input type="checkbox"/>	1140	583
<input type="checkbox"/>	1210	43
<input type="checkbox"/>	1310	72
<input type="checkbox"/>	1360	265
<input type="checkbox"/>	1390	107
<input type="checkbox"/>	1410	601
<input type="checkbox"/>	1430	13

Console Check all With selected: Edit Copy Delete Export

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

Query

```
CREATE VIEW ventas2001 AS
SELECT m.clave, SUM(e.cantidad) AS 'Total'
FROM entregan e, materiales m
WHERE e.clave = m.clave
AND e.fecha BETWEEN '2001/01/01' AND '2001/12/31'
GROUP BY m.clave;
```

```
SELECT clave FROM ventas2001
ORDER BY Total DESC
LIMIT 1
```

The screenshot shows the phpMyAdmin interface. On the left, the database structure is displayed under the 'entregas' database. The 'Tables' section contains 'New', 'entregan', 'materiales', 'proveedores', and 'proyectos'. The 'proveedores' table has columns 'razonsocial' (varchar) and 'rfc' (PRI, varchar). The 'Views' section contains 'New', 'importeventas', 'razonsocialav1', 'razonsocialav2', 'sortedprojects', 'threeigitsandsix', and 'ventas2001'. On the right, the 'SQL' tab is active, showing the query: `SELECT clave FROM ventas2001 ORDER BY Total DESC LIMIT 1;`. The results show one row: 'clave' 1260. Below the results, there are options to 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'. A 'Bookmark this SQL query' section is also present.

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

Query

```
SELECT m.descripcion  
FROM materiales m  
WHERE M.descripcion LIKE '%ub%'
```

Showing rows 0 - 11 (12 total, Query took 0.0005 seconds.)

```
/*Productos que contienen el patrón 'ub' en su nombre.*/
SELECT m.descripcion FROM materiales m WHERE M.descripcion LIKE '%ub%';
```

	descripcion
<input type="checkbox"/>	Recubrimiento P1001
<input type="checkbox"/>	Recubrimiento P1010
<input type="checkbox"/>	Recubrimiento P1019
<input type="checkbox"/>	Recubrimiento P1028
<input type="checkbox"/>	Recubrimiento P1037
<input type="checkbox"/>	Tubería 3.5
<input type="checkbox"/>	Tubería 4.3
<input type="checkbox"/>	Tubería 3.6
<input type="checkbox"/>	Tubería 4.4
<input type="checkbox"/>	Tubería 3.7
<input type="checkbox"/>	Tubería 4.5
<input type="checkbox"/>	Tubería 3.8

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

Query

```

SELECT p.denominacion, SUM(e.cantidad * m.precio) AS 'Total a pagar'
FROM proyectos p, entregan e, materiales m
WHERE e.numero = p.numero
AND e.clave = m.clave
GROUP BY p.denominacion
ORDER BY SUM(e.cantidad * m.precio) DESC;

```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'entregas'. The current table is 'proyectos'. The left sidebar shows the database structure with tables 'materiales', 'proveedores', and 'proyectos'. The 'proyectos' table is selected, showing its columns: 'denominacion' (varchar), 'numero' (PRI, int), and 'rfc' (PRI, varchar). A query is displayed in the top right:

```
/* Denominación y suma del total a pagar para todos los proyectos. */
SELECT p.denominacion, SUM(e.cantidad * m.precio) AS 'Total'
FROM entregas e, materiales m
WHERE e.numero = p.numero AND e.clave = m.clave
GROUP BY p.denominacion;
```

The results of the query are listed below, showing the project name and the total amount to be paid:

denominacion	Total a pagar
Ampliación de la carretera a la huasteca	565135
Aztecon	146595
CIT Campeche	157755
CIT Yucatan	851700
Construcción de Hospital Infantil	144295
Construcción de plaza Magnolias	122969
Disco Atlantic	158100
Educando en Coahuila	620610
Infonavit Durango	321135
Mexico sin ti no estamos completos	260290
Queretaro limpio	449715
Reconstrucción del templo de Guadalupe	220580

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).

Query

```
CREATE VIEW televisa AS SELECT pry.denominacion, pr.rfc, pr.razonsocial
FROM proyectos pry, proveedores pr, entregan e
WHERE pry.numero = e.numero
AND pr.rfc = e.rfc
AND pry.denominacion = 'Televisa en acción'
GROUP BY pr.rfc;
```

```

CREATE VIEW coahuila AS SELECT pry.denominacion, pr.rfc, pr.razonsocial
FROM proyectos pry, proveedores pr, entregan e
WHERE pry.numero = e.numero
AND pr.rfc = e.rfc
AND pry.denominacion = 'Educando en Coahuila'
GROUP BY pr.rfc;

```

```

SELECT *
FROM televisa
WHERE rfc NOT IN (SELECT rfc
                   FROM coahuila);

```

The screenshot shows the phpMyAdmin interface with the following details:

- Left Sidebar (Database Structure):**
 - Entregas (Tables: New, entregan, materiales, proveedores, proyectos)
 - Views (New, coahuila, importeventas, razonsociallav1, razonsociallav2, sortedprojects, televisa, threedigitsandsix, ventas2001)
 - information_schema (Tables: New, mensaje, privilegios, roles, rol_privilegio, usuarios, usuario_rol)
- Top Bar:** Server: 127.0.0.1 > Database: entregas > View: televisa. Buttons: Browse, Structure, SQL, Search, Export, Privileges, Operations, Tracking.
- Query Results:**
 - Message: Showing rows 0 - 1 (2 total, Query took 0.0015 seconds.)
 - SQL Query: SELECT * FROM televisa WHERE rfc NOT IN (SELECT rfc FROM coahuila);
 - Table Data:

	denominacion	rfc	razonsocial
1	Televisa en acción	CCCC800101	La Ferre
2	Televisa en acción	DDDD800101	Cecoferre
 - Buttons: Show all, Number of rows: 25, Filter rows: Search this table.
 - Extra options: Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh].
 - Query results operations: Print, Copy to clipboard, Export, Display chart, Create view.
 - Bookmark this SQL query: Label: _____, Let every user access this bookmark.

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).

Query

```
SELECT pry.denominacion, pr.rfc, pr.razonsocial  
FROM proyectos pry, proveedores pr, entregan e  
WHERE pry.numero = e.numero  
AND pr.rfc = e.rfc  
AND pr.rfc IN (SELECT pr.rfc FROM proyectos pry, proveedores pr, entregan e  
WHERE pry.numero = e.numero  
AND pr.rfc = e.rfc  
AND pry.denominacion = 'Televisa en acción'  
GROUP BY pr.rfc)  
AND pr.rfc NOT IN (SELECT pr.rfc FROM proyectos pry, proveedores pr, entregan e  
WHERE pry.numero = e.numero  
AND pr.rfc = e.rfc  
AND pry.denominacion = 'Educando en Coahuila'  
GROUP BY pr.rfc)  
GROUP BY pr.rfc;
```

phpMyAdmin

Server 127.0.0.1 > Database entregas > View: televisa

Browse Structure SQL Search Export Privileges Operations Tracking

Delimiter : Show this query here again Retain query box Rollback when finished Enable foreign key checks Go

Hide query box

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 1 (2 total, Query took 0.0026 seconds.)

```
SELECT pry.denominacion, pr.rfc, pr.razonsocial FROM proyectos pry, proveedores pr, entregan e WHERE pry.numero = e.numero AND pr. (SELECT pr.rfc FROM proyectos pry, proveedores pr, entregan e WHERE pry.numero = e.numero AND pr.rfc = e.rfc AND pry.denominacion BY pr.rfc) AND pr.rfc NOT IN (SELECT pr.rfc FROM proyectos pry, proveedores pr, entregan e WHERE pry.numero = e.numero AND pr.rfc 'Educando en Coahuila' GROUP BY pr.rfc) GROUP BY pr.rfc;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

Extra options

denominacion	rfc	razonsocial
CIT Campeche	CCCC800101	La Ferre
Mexico sin ti no estamos completos	DDDD800101	Cecoferre

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

Query results operations

Print Copy to clipboard Export Display chart Create view

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.precio, m.descripcion
FROM proyectos pry, materiales m, entregan e, proveedores p
WHERE m.clave = e.clave
AND e.rfc = p.rfc
AND e.numero = pry.numero
AND pry.denominacion = 'Televisa en acción'
AND e.rfc IN (SELECT e.rfc FROM entregan e, proyectos p WHERE p.numero = e.numero
AND p.denominacion = 'Educando en Coahuila')

```

The screenshot shows the phpMyAdmin interface for a database named 'entregas'. On the left, the database structure is displayed with tables like 'entregas', 'materiales', 'proveedores', 'proyectos', and various views. The 'materiales' table is currently selected. The main area shows a query result for a specific SQL query:

```

SELECT m.precio, m.descripcion FROM proyectos pry, materiales m, entregan e, proveedores p WHERE m.clave = e.clave AND e.rfc = p.rfc AND e.numero = pry.numero
AND pry.denominacion = 'Televisa en acción' AND e.rfc IN (SELECT e.rfc FROM entregan e, proyectos p WHERE p.numero = e.numero AND p.denominacion = 'Educando en
Coahuila') GROUP BY e.rfc;

```

The result table displays two rows of data:

	precio	descripcion
	50	Ladrillos rojos
	34	Tepetate

Below the table are standard MySQL query operations: Print, Copy to clipboard, Export, Display chart, and Create view.

Reto: Usa solo el operador NOT IN en la consulta anterior (No es parte de la entrega).

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

```

SELECT m.descripcion, COUNT(e.clave) AS 'Numero de entregas', SUM(m.precio *
e.cantidad) AS 'Costo total'

FROM materiales m, entregan e

WHERE m.clave = e.clave

GROUP BY (m.descripcion)

ORDER BY SUM(m.precio * e.cantidad) DESC

```

phpMyAdmin

Server 127.0.0.1 » Database entregas » Table: materiales

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers Hide query box

Showing rows 0 - 24 (42 total, Query took 0.0010 seconds)

```
/* Nombre del material, cantidad de veces entregados y total del costo de dichas entregas por material de todos los proyectos. */
SELECT m.descripcion, COUNT(e.clave) AS 'Número de entregas', SUM(m.precio * e.cantidad) AS 'Costo total' FROM materiales m, entregan e WHERE m.clave = e.clave GROUP BY (m.descripcion) ORDER BY SUM(m.precio * e.cantidad) DESC;
```

Edit inline | Edit | Create PHP code |

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

descripcion	Número de entregas	Costo total
Cantera gris	2	1102310
Pintura C1011	4	748825
Cantera rosa	2	576520
Tubería 4.4	2	255530
Cemento	2	252600
Tubería 4.5	2	249500
Pintura B1021	4	234000
Varilla 4/34	2	197050
Varilla 3/19	2	193040
Recubrimiento P1037	2	190960
Recubrimiento P1019	2	182880
Cantera blanca	2	160400
Tubería 3.7	2	155280
Consola P1001	2	148200

New Tables entregan materiales proveedores proyectos Views coahuila importeventas razonsociallav1 razonsociallav2 sortedprojects televisa threedigitsandsix