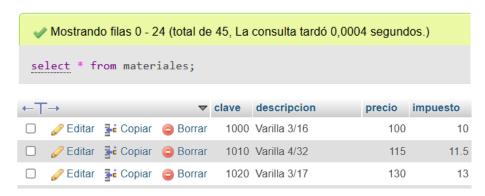
Laboratorio 20 - TC2005B

Consultas en SQL

Consulta de una tabla completa



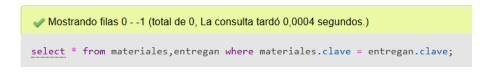
Selección



Proyección



Reunión Natural



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				1010	55555555	50.40	1007.00.00	500

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

No, solo se muestran los registros que están tanto en la tabla de materiales como en la tabla de entregan, por lo que registros de materiales no entregados no aparecerían en la consulta

Reunión con criterio específico



Unión

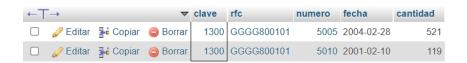


clave	rfc	numero	fecha	cantidad
1300	GGGG800101	5005	2004-02-28	521
1300	GGGG800101	5010	2001-02-10	119

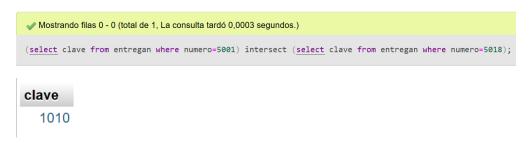
¿Cuál sería una consulta que obtuviera el mismo resultado sin usar el operador Unión? Compruébalo.

Podemos utilizar el operador 'IN' para indicar que la clave sea 1450 y 1300 de la tabla entregan, como no hay registros con la clave 1450 solo muestra resultados con la clave 1300.

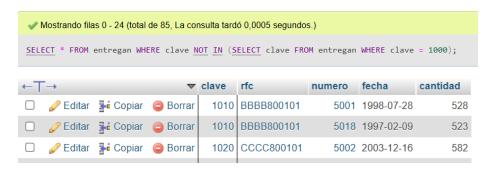
```
✓ Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0003 segundos.)
SELECT * FROM entregan WHERE clave IN (1450, 1300);
```



Intersección



Diferencia



Producto cartesiano



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

El producto cartesiano combina cada fila de la tabla entregan con cada fila de la tabla materiales

Total de tuplas = tuplas_entregan x tuplas_materiales

Construcción de consultas a partir de una especificación

Consulta para obtener las descripciones de los materiales entregados en el año 2000.

¿Por qué aparecen varias veces algunas descripciones de material?

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?

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.

Uso de expresiones

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

Operadores de cadena



¿Qué resultado obtienes?

Obtiene todos los materiales en cuya descripción se encuentre la palabra si

Explica que hace el símbolo '%'.

El símbolo '%' en esta consulta nos indica que obtiene los materiales cuya descripción empieza con la palabra 'Si' y siga con cualquier secuencia de caracteres será seleccionada

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

Solo obtendría los materiales cuya descripción es 'Si'

¿Qué resultado obtienes?

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

SELECT * FROM materiales WHERE Descripcion LIKE 'Si';

Clave descripcion precio impuesto
```

Explica a qué se debe este comportamiento.

Solo se seleccionarán las filas donde la descripción sea exactamente igual a "Si", sin que haya ningún otro carácter antes ni después, como no hay ningún material cuya descripción sea únicamente 'Si' no selecciona nada.

Operador 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 (Apellido + ', ' + Nombre) as Nombre FROM Personas;

DECLARE @foo varchar(40);
DECLARE @bar varchar(40);
```

```
DECLARE @bar varchar(40);

SET @foo = '¿Que resultado';

SET @bar = '¿¿¿???' '

SET @foo += ' obtienes?';

PRINT @foo + @bar;
```

Código equivalente en MySQL:

SELECT CONCAT(@foo, @bar) AS Resultado;

```
SELECT CONCAT(Apellido, ', ', Nombre) AS Nombre FROM Personas;

SET @foo = '¿Que resultado';

SET @bar = ' ¿¿¿???';

SET @foo = CONCAT(@foo, ' obtienes?');
```

¿Para qué sirve DECLARE?

En sql DECLARE nos sirve para definir variables que nos permiten asignar y recuperar valores.

MySQL no soporta directamente DECLARE y este solo puede ser utilizado dentro de procedimientos almacenados.

¿Cuál es la función de @foo?

@foo es el identificador que se usa para definir una variable que almacena un valor temporal

¿Que realiza el operador SET?

SET asigna los valores a las variables

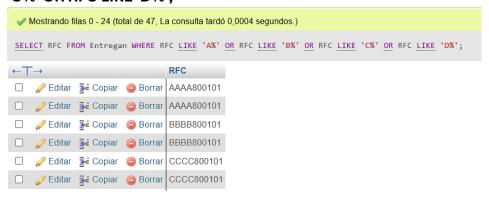
Explica el comportamiento, función y resultado de cada una de las siguientes consultas:

```
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';
```

Primero es importante mencionar que estas consultas están hechas para SQL server por lo que no funcionan en MySQL, en MySQL los corchetes no funcionan de la misma manera y normalmente se utiliza el operador LIKE junto con el comodín % para obtener los resultados.

Las consultas modificadas a MYSQL quedarán como:

SELECT RFC FROM Entregan WHERE RFC LIKE 'A%' OR RFC LIKE 'B%' OR RFC LIKE 'C%' OR RFC LIKE 'D%';



Se seleccionan los RFC que comiencen con cualquier letra de la 'A' a la 'D'. El % nos indica que puede haber cualquier cantidad de caracteres después de la letra inicial.

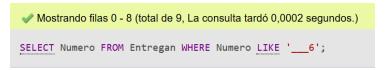
SELECT RFC FROM Entregan WHERE RFC NOT LIKE 'A%';

```
✓ Mostrando filas 0 - 24 (total de 75, La consulta tardó 0,0002 segundos.)
SELECT RFC FROM Entregan WHERE RFC NOT LIKE 'A%';
```



Se seleccionan los RFC que no empiecen con la letra A al igual que en la consulta anterior el % nos permite cualquier cantidad de caracteres después de ese primer carácter

SELECT Numero FROM Entregan WHERE Numero LIKE '___6';





Selecciona los números de entrega que contienen 4 caracteres (ya que el guion bajo se repite 4 veces) y terminan con un 6

Operadores compuestos.

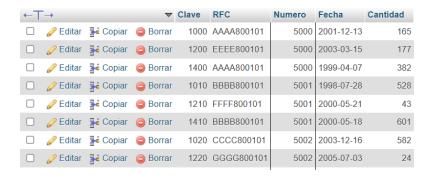
Los operadores compuestos ejecutan una operación y establecen un valor.

- + = (Suma igual)
- = (Restar igual)
- * = (Multiplicar igual)
- / = (Dividir igual)
- % = (Módulo igual)

Operadores Lógicos.

```
✓ Mostrando filas 0 - 24 (total de 43, La consulta tardó 0,0004 segundos.)

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



¿Cómo filtrarías rangos de fechas?

Al igual que lo hicimos con los números solo que esta vez especificando un rango de fechas deseado, por ejemplo, entre 1999 y 2001:



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





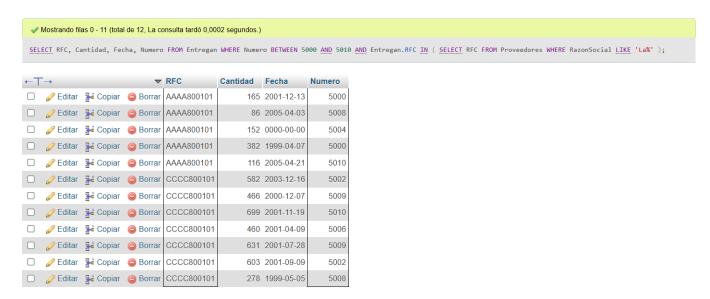
¿Qué hace la consulta?

En la consulta se hacen diferentes filtros para obtener la información que requerimos. Seleccionamos el rfc, cantidad, fecha y numero de las entregas cuyo numero de entrega esta entre 5000 y 5010 y existe al menos un proveedor cuyo RFC coincida con el RFC de entregan y cuya razón social comience con "La".

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

Se utiliza para definir una subconsulta

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



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.

Obtener los registros de la tabla Entregan donde la Cantidad de un material sea mayor que todas las cantidades de entregas con el Numero = 1000.





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

¿Qué hace la siguiente sentencia? Explica por qué. SELECT TOP 2 * FROM Proyectos

Va a seleccionar los dos primeros registros de la tabla "Proyectos", ya que el operador Top en SQL server se utiliza para limitar el numero de filas que se muestran.

¿Qué sucede con la siguiente consulta? Explica por qué. SELECT TOP Numero FROM Proyectos

Como el operador Top no tiene especificado ningún parámetro es posible que no podamos ejecutar la consulta ya que Top no acepta una columna como parámetro.

Modificando la estructura de una 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:

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

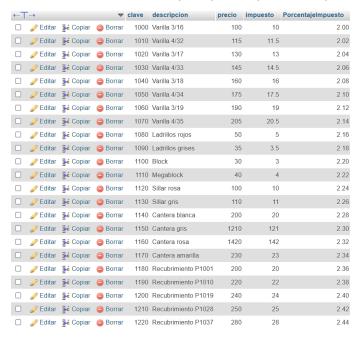
ALTER TABLE Materiales ADD PorcentajeImpuesto NUMERIC(6, 2);
```

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.

```
✓ 45 filas afectadas. (La consulta tardó 0,0003 segundos.)

<a href="https://www.eps.com/uppares/">UPDATE</a> Materiales <a href="https://www.eps.com/uppares/">SET</a> PorcentajeImpuesto = 2 * Clave / 1000;
```

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?



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!

Crea vistas para cinco de las consultas que planteaste anteriormente en la práctica. Posteriormente revisa cada vista creada para comprobar que devuelve el mismo resultado.

MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0004 segundos.)

CREATE VIEW vista_materiales_mexico_sin_ti AS SELECT M.Clave, M.Descripcion FROM Materiales M JOIN Entregan E ON M.Clave = E.Clave JOIN Proyectos P ON E.Numero = P.Numero where P.Denominacion = 'México sin ti no estamos completos';