Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Querétaro



TC2005B Construcción de software y toma de decisiones

Ricardo Cortés Espinosa

Laboratorio 20: Consultas en SQL

Marco Antonio Camalich Pérez

A01351725

Fecha de entrega: 17/03/2022

Abre una sesión de Analizador de Consultas y ejecuta cada una de las sentencias SQL. 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.

Utilizando el esquema relacional.

Materiales (Clave, Descripcion, Precio)

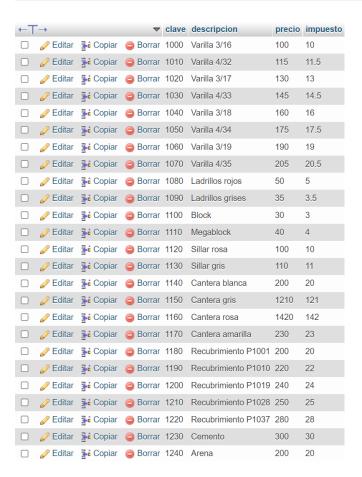
Proveedores (RFC, RazonSocial)

Proyectos (Numero, Denominacion)

Entregan(Clave, RFC, Numero, Fecha, Cantidad)

Consulta de un tabla completa

✓ Mostrando filas 0 - 24 (total de 45, La consulta tardó 0,0015 segundos.)
select * from materiales;



Selección

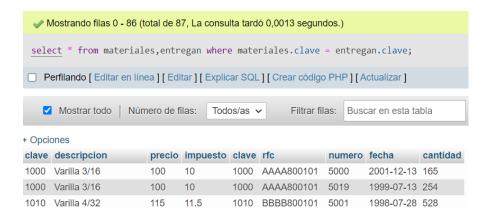


Proyección

```
✓ Mostrando filas 0 - 24 (total de 87, La consulta tardó 0,0012 segundos.)
select clave,rfc,fecha from entregan;
```



Reunión Natural



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

Si aparecería, ya que la reunión natural unifica las tablas en cuestión sin hacer distinción de intersecciones entre ambas.

Reunión con criterio específico



Unión (se ilustra junto con selección)



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

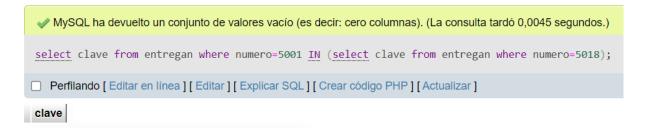
SELECT*

FROM Entregan

Where Clave=1450 OR Clave=1300



Intersección (se ilustra junto con selección y proyección)



Diferencia (se ilustra con selección)



Producto cartesiano



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

En entregan hay 87 tuplas y en materiales hay 45. El producto cartesiano relaciona cada tupla de una tabla con el total de tuplas existentes en la otra, por lo que se realiza la multiplicación de 87*45 y eso da las 3915 filas que se muestran en el gestor.

Construcción de consultas a partir de una especificación

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



¿Por qué aparecen varias veces algunas descripciones de material?

Porque la relación con la tabla "Entregan" da a entender que se realizó más de una entrega del material descrito en el año 2000.

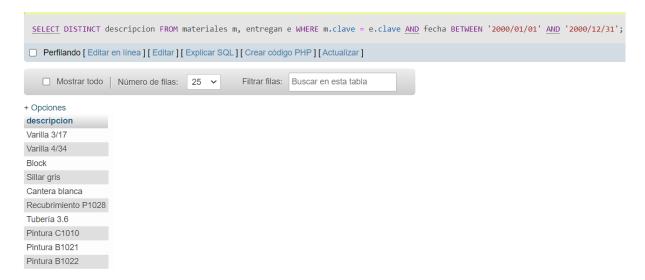
Uso del calificador distinct

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.

¿Qué resultado obtienes en esta ocasión?

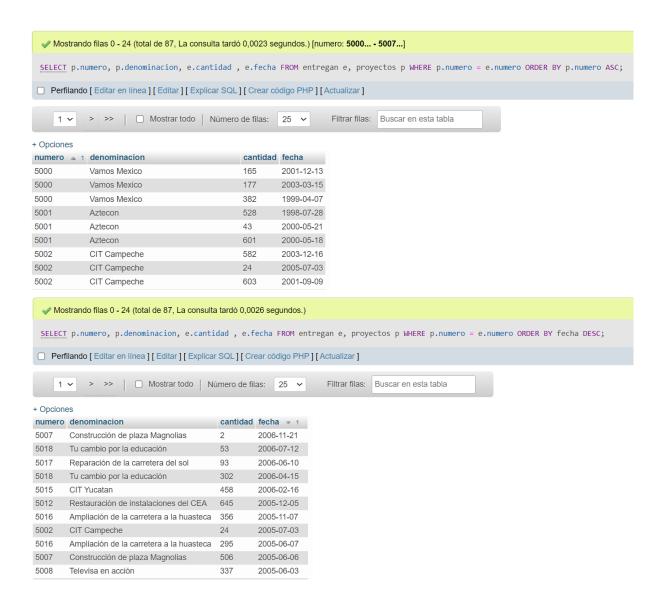
Se eliminan los valores repetidos de la búsqueda anterior.



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.

Ya que la consulta es ambigua al final, se presentan los scripts del ordenamiento por número de proyecto y de la fecha más reciente a la más antigua respectivamente.



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



¿Qué resultado obtienes?

Las descripciones de los materiales cuyo string contiene 'Si' al inicio.

Explica que hace el símbolo '%'.

Buscar coincidencias de cadena dentro de los registros para regresar las tuplas que cumplan con esa fracción de string. Si se coloca antes del fragmento, buscará que este se encuentre al final del registro, si este se coloca después del fragmento, buscará que el pedazo se encuentre al inicio del registro. Finalmente, si se coloca en ambos lados del fragmento, buscará que existan caracteres antes y/o después de este. ('%Si', 'Si%' y '%Si%' respectivamente.)

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

No devolverá nada como respuesta ya que ahora se buscará 'Si' como el total del registro y no como parte del string.

¿Qué resultado obtienes?

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0013 segundos.)
SELECT * FROM materiales where Descripcion LIKE 'Si';
Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]
clave descripcion precio impuesto
```

Explica a qué se debe este comportamiento.

El Operador '%' refiere el 'Si' como parte del string, por lo que 'Silla' o 'Sillón' cumplen con lo solicitado. Pero si se evade el %, se buscará el 'Si' como el total del string, por lo que sí 'Si' no existe en ninguna descripción sin compañía de otro carácter, no habrá un conjunto de respuestas válidas para devolver.

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

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

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0018 segundos.)
SELECT RFC FROM Entregan WHERE RFC LIKE '[A-D]%';
Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]
RFC
```

La consulta busca desplegar los RFCs de la tabla 'Entregan' donde estos registros contienen como parte del string, no como la totalidad del mismo, la coincidencia de letras del intérvalo de la letra A a la D, para posteriormente manejar otros caracteres adelante. Dado a que no hay tuplas que cumplan con la condición anterior, el resultado es la columna RFC con valores vacíos.

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

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0013 segundos.)
SELECT RFC FROM Entregan WHERE RFC LIKE '[^A]%';
Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]
RFC
```

La consulta busca desplegar los RFCs de la tabla 'Entregan' donde estos registros no contienen dentro del string, el intérvalo que solo maneja la letra A, para posteriormente manejar otros caracteres adelante. Dado a que no hay tuplas que cumplan con la condición anterior, el resultado es la columna RFC con valores vacíos.

SELECT Numero FROM Entregan WHERE Numero LIKE '___6';



Se muestran 8 filas que provienen de la selección de la columna número de la tabla Entregan donde el número posee 3 espacios indefinidos previo a un cuarto lugar ocupado con un 6. El resultado regresa números como 5016 o 5006 que poseen tres números aleatorios antes que el 6 del final.

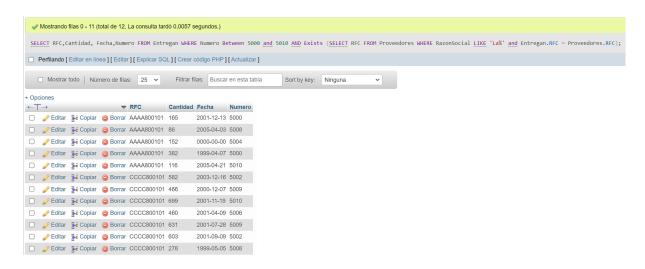
BETWEEN

¿Cómo filtrarías rangos de fechas?



Como se realizó en el inciso del año 2000, se coloca el primer día del año y el último dentro de un BETWEEN con su respectivo AND, esto delimita rangos de días que arrojan resultados basados en las fechas de los registros.

EXISTS



¿Qué hace la consulta?

La consulta requiere de una subconsulta para obtener el RFC, Cantidad, Fecha y Número partiendo de la tabla Entregan con la condición de que el número de la entrega se encuentre entre 5000 y 5010. Lo que realiza la subconsulta es observar la existencia de las tuplas 'RFC' dentro de la subconsulta que especifica que se contrastan los proveedores que poseen 'La' dentro de su razón social (Como es el caso de 'La Fragua') para posteriormente hacer la igualación necesaria entre el RFC de 'Entregan' y el de 'Proveedores' para que la consulta general funcione. Esto da de resultado las entregas de los proveedores con 'La' en su Razón Social y dentro del rango de 5000 y 5010 previamente especificado.

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

Definir la subconsulta ya que se necesita encerrar el query como un todo para poder contrastar la existencias de las filas de una consulta dentro de otra.

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

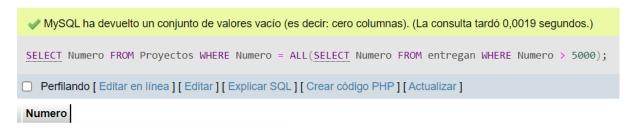
SELECT RFC, Cantidad, Fecha, Numero FROM Entregan WHERE Numero Between 5000 and 5010 AND RFC NOT IN (SELECT RFC FROM Proveedores WHERE RazonSocial NOT LIKE 'La%' and Entregan.RFC = Proveedores.RFC);



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

SELECT Numero FROM Proyectos WHERE Numero = ALL(SELECT Numero FROM entregan WHERE Numero > 5000);

El ejemplo superior busca listar los números de la tabla proyecto, solamente si todos los números de las entregas son mayores a 5000, sin contar ese mismo valor. Dado a que no se cumple con esa condición, el resultado del query está vacío.



¿Qué hacen las siguientes sentencias? Explica por qué.

SELECT TOP 2 * FROM Proyectos



El equivalente a Top en MYSQL es LIMIT, estos lo que hacen es delimitar la cantidad de tuplas que se imprimen por lo que limitarlo a 2, regresa las primeras 2 tuplas de la tabla entera de proyectos.

SELECT TOP Numero FROM Proyectos



Marca un error porque numero es un atributo de la tabla Proyectos que no tiene un valor numérico como tal, es una columna que posee diferentes valores y no maneja un dato comparable para acompañar al LIMIT en si.

Modificando la estructura de un tabla existente.

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0127 segundos.)
ALTER TABLE materiales ADD PorcentajeImpuesto NUMERIC(6,2);
[Editar en línea] [Editar] [Crear código PHP]
```

¿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?



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.

- CREATE VIEW vista1 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);
- CREATE VIEW vista2 AS SELECT Numero FROM Proyectos WHERE Numero = ALL(SELECT Numero FROM entregan WHERE Numero > 5000);
- 3. CREATE VIEW vista3 AS SELECT * FROM Proyectos Limit 2
- 4. CREATE VIEW vista4 AS SELECT * FROM materiales
- 5. CREATE VIEW vista5 AS SELECT * FROM Entregan WHERE Clave=1450 OR Clave=1300

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".

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



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

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

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0028 segundos.)
SELECT m.clave, m.descripcion FROM materiales m, proveedores p, entregan e WHERE e.clave = m.clave AND e.rfc = p.rfc AND p.razonsocial="Acme tools";
□ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]
clave descripcion
```

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

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



El Total entregado por cada material en el año 2000.

SELECT clave, SUM(cantidad) as 'Total entregado por material' FROM entregan WHERE fecha BETWEEN '2000/01/01' AND '2000/12/31' GROUP BY clave;



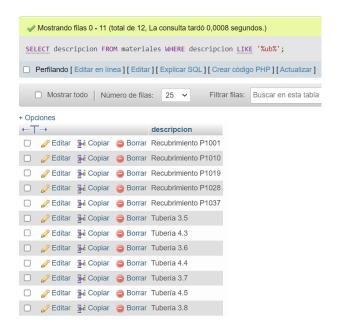
La Clave del material más vendido durante el 2001. (se recomienda usar una vista intermedia para su solución)

SELECT clave, SUM(cantidad) as 'Venta del material' FROM entregan WHERE fecha BETWEEN '2001/01/01' AND '2001/12/31' GROUP BY clave ORDER BY SUM(cantidad) DESC LIMIT 1;



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 p.denominacion, SUM(e.Cantidad * (m.precio + m.impuesto)) as 'Total a
Pagar' FROM proyectos p, entregan e, materiales m Where p.numero = e.numero
AND m.clave = e.clave GROUP BY p.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 proveedoresTelevisa as SELECT Denominacion, PR.RFC, razonsocial FROM Proyectos P , proveedores PR, entregan E Where E.numero = P.numero AND E.rfc = PR.rfc AND P.denominacion = 'Televisa en acción';

✓ MySQL ha devuelto un conjunto de valores vacio (es dedir. cero columnas). (La consulta tardó 0,0073 segundos.)

create view proveedores Televisa as SELECT. Denominación, PR.RFC, razonsocial FROM Proyectos P , proveedores PR, entregan E where E.numero = P.numero AND E.rfc = PR.rfc AND P.denominación = 'Televisa en acción';

CREATE VIEW proveedoresEducandoCoahuila as SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero = P.numero AND E.rfc = PR.rfc AND P.denominacion = 'Educando en Coahuila';

✓ MySQL ha devuelto un conjunto de valores vacio (es decir: cero columnas). (La consulta tardó 0,0055 segundos.)

CREATE VIEM proveedores EducandoCoahuila as SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero * P.numero * MD E.rfc * PR.rfc * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM proveedores EducandoCoahuila as SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero * P.numero * MD E.rfc * PR.rfc * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM proveedores EducandoCoahuila as SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero * P.numero * MD E.rfc * PR.rfc * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM proveedores EducandoCoahuila as SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero * P.numero * MD E.rfc * PR.rfc * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM proveedores EducandoCoahuila as SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero * P.numero * MD E.rfc * PR.rfc * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREATE VIEM PROVEED * MD P.denominacion * 'Educando en Coahuila';

CREA

SELECT t.Denominacion, t.RFC, t.razonsocial FROM proveedorestelevisa as t, proveedoreseducandocoahuila as c WHERE t.RFC NOT IN (c.RFC) GROUP BY t.razonsocial;



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 Denominacion, pr.RFC, razonsocial FROM Proyectos p, proveedores pr, entregan e Where e.numero = p.numero AND e.rfc = pr.rfc AND p.denominacion = 'Televisa en acción' AND pr.rfc NOT IN (SELECT pr.RFC FROM Proyectos p, proveedores pr, entregan e Where e.numero = p.numero AND e.rfc = pr.rfc AND p.denominacion = 'Educando en Coahuila') GROUP BY razonsocial;



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.descripcion, (precio + impuesto) as 'Costo del material' FROM Proyectos P , proveedores PR, entregan E, materiales M WHERE E.numero = P.numero AND E.rfc = PR.rfc AND E.clave = M.clave AND P.denominacion = 'Televisa en acción' AND PR.rfc IN (SELECT PR.RFC FROM Proyectos P , proveedores PR, entregan E Where E.numero = P.numero AND E.rfc = PR.rfc AND P.denominacion = 'Educando en Coahuila') GROUP BY M.descripcion;

SELECT M.descripcion, (precio + impuesto) as 'Costo del material' FROM Proyectos P , proveedores PR, entregan E, materiales M MHERE E.numero = P.numero AND E.rfc = PR.rfc AND E.clave = M.clave = M
□ Perfilando [Editar en linea] [Editar] [Explicar SQL.] [Crear código PHP.] [Actualizar]
☐ Mostrar todo │ Número de filas: 25 ✔ Filtrar filas: Buscar en esta tabla
+ Opciones
← T→ ▼ descripcion Costo del material
□ dEditar ¼ Copiar ♦ Borrar Ladrillos rojos 55
Folitar 3 conjar @ Borrar Tenetate 37 40000009536743

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

SELECT denominacion, descripcion, COUNT(*)as 'Cantidad de veces entregado', (cantidad*(precio+impuesto)) as 'Total' FROM materiales M, proyectos P, entregan E WHERE M.clave = E.clave AND P.numero = E.numero GROUP BY denominacion, descripcion;

