

Si algún material no se ha entregado ¿Aparecería en el resultado de esta consulta?
No aparecería, en entregan nada más están los que sí.

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

Compruébalo

```
SELECT *
```

```
FROM entregan
```

```
WHERE clave=1300 OR clave=1430;
```

	Clave	RFC	Numero	Fecha	Cantidad
	1300	GGGG800101	5005	10/06/02	521
	1300	GGGG800101	5005	2/02/03	457
	1300	GGGG800101	5010	8/01/03	119
	1430	DDDD800101	5003	2/09/99	576
	1430	DDDD800101	5007	9/01/98	13
	1430	DDDD800101	5007	10/06/02	506
*	0		0		0

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

Las tuplas se suman y aparecen todas las permutaciones posibles.

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

```
SELECT Descripcion
```

```
FROM Materiales AS M, Entregan AS E
```

```
WHERE M.clave = E.clave AND
```

```
E.fecha BETWEEN #01/01/2000# AND #31/12/2000#;
```

Descripción ▾
Varilla 3/16
Varilla 4/32
Varilla 4/32
Varilla 4/33
Varilla 3/18
Varilla 4/34
Varilla 3/19
Varilla 3/19
Varilla 3/19
Varilla 4/35
Ladrillos grises
Block
Megablock
Cantera rosa
Recubrimiento
Recubrimiento
Recubrimiento
Recubrimiento
Arena
Tepetate
Tubería 3.6
Tubería 4.4
Tubería 3.7
Pintura C1010
Pintura B1020
Pintura B1020
Pintura B1021
Pintura C1012

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

```
SELECT DISTINCT Descripción
FROM Materiales AS M, Entregas AS E
WHERE M.clave = E.clave AND
E.fecha BETWEEN #01/01/2000# AND #31/12/2000#;
```

Descripción ▾
Arena
Block
Cantera rosa
Ladrillos griseo
Megablock
Pintura B1020
Pintura B1021
Pintura C1010
Pintura C1012
Recubrimiento
Recubrimiento
Tepetate
Tubería 3.6
Tubería 3.7
Tubería 4.4
Varilla 3/16
Varilla 3/18
Varilla 3/19
Varilla 4/32
Varilla 4/33
Varilla 4/34
Varilla 4/35

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.

numero	denominaci	fecha	cantidad
5004	Educando en C	1/09/03	270
5011	Remodelación	5/08/03	549
5011	Remodelación	28/06/03	368
5011	Remodelación	1/06/03	429
5016	Ampliación de	7/03/03	622
5009	Disco Atlantic	10/02/03	2
5005	Infonavit Dura	2/02/03	457
5002	CIT Campeche	1/02/03	24
5004	Educando en C	12/01/03	152
5019	Queretaro lim	12/01/03	107
5010	Construcción d	8/01/03	119
5003	Mexico sin ti n	6/01/03	530
5012	Restauración d	6/01/03	698
5012	Restauración d	9/12/02	312
5007	Construcción d	7/12/02	448
5014	Remodelación	2/12/02	324
5011	Remodelación	7/11/02	699
5009	Disco Atlantic	5/11/02	461
5019	Queretaro lim	2/10/02	199
5008	Televisa en acc	29/07/02	107
5015	CIT Yucatan	12/07/02	540
5005	Infonavit Dura	8/07/02	71
5006	Reconstrucción	6/07/02	562
5013	Reparación de	4/07/02	658
5018	Tu cambio por	14/06/02	407
5007	Construcción d	10/06/02	506
5005	Infonavit Dura	10/06/02	521
5013	Reparación de	5/06/02	423
5009	Disco Atlantic	7/05/02	523
5010	Construcción d	3/05/02	441
5009	Disco Atlantic	3/05/02	467
5011	Remodelación	12/04/02	72
5008	Televisa en acc	12/04/02	324
5014	Remodelación	7/04/02	265
5010	Construcción d	4/04/02	691
5013	Reparación de	4/04/02	63
5018	Tu cambio por	29/03/02	523
5000	Vamos Mexico	12/03/02	382
5018	Tu cambio por	3/03/02	302
5019	Queretaro lim	8/02/02	665
5011	Remodelación	6/01/02	308

Modifique el ejemplo de la lectura para que se adecue a la base de datos que se tiene.
SELECT * FROM proyectos where denominacion LIKE 'Re%';

¿Qué resultado obtienes?

Todos los proyectos en los que su denominación empieza con “Re”.

Explica que hace el símbolo '%'

Es para especificar que no importa qué es lo que haya después, pues sin él nada buscaría denominaciones llamadas “Re”.

```
SELECT * FROM proyectos where denominacion LIKE 'Re';
```

¿Qué resultado obtienes? Nada.

Explica a qué se debe este comportamiento. Como mencioné, sin el % busca denominaciones que específicamente se llaman “Re”.

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

¿Que resultado obtienes? ¿¿¿???

¿Para qué sirve DECLARE? Para crear variables en SQL.

¿Cuál es la función de @foo? Ser la variable donde se guarda “¿Que resultado” y “obtiene?”

¿Qué realiza el operador SET? Actualizar las variables con valores nuevos.

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

Busca el RFC incluyendo de A-D, y la segunda es que empiecen con A.

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

Busca registros en Entregan de 4 dígitos que acaben en 6.

¿Cómo filtrarías rangos de fechas? Con Between,

```
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?

Registros cuya razón social empieza con “La” y el número de entrega esté entre 5000 y 5010

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

Es para hacer una subconsulta. Esto permite hacer una tabla temporal con el RFC de proveedores que empiecen con "La".

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);
```

	RFC	Cantidad	Fecha	Numero
*		0		0

En mi caso no hubo suficientes datos para regresar una consulta completa.

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);
```

¿Qué hace la siguiente sentencia? Explica por qué.

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

Usa la palabra reservada TOP para regresar los primeros dos registros de la tabla Proyectos.

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

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

Es incorrecta, pues TOP debe de ser escrito seguido por un número, el cual indicará el número de registros que regresará.

Modifica la tabla de materiales y le agrega columna PorcentajeImpuesto Numérica.

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

¿Qué consulta usarías para obtener el importe de las entregas es decir, el total en dinero de lo entregado, basado en la cantidad de la entrega y el precio del material y el impuesto asignado?

```
SELECT E.RFC, SUM((precio + impuesto) * cantidad) AS "Total  
entregado"  
FROM Materiales AS M, Entregan AS E  
WHERE M.clave = E.clave  
GROUP BY E.RFC;
```

Consulta #1:

```
CREATE VIEW Proveedores_QueEmpiezancon_La  
AS SELECT Entregan.RFC, Entregan.Cantidad, Entregan.Fecha,  
Entregan.numero  
FROM Proveedores, Entregan  
WHERE RazonSocial LIKE 'La%' and Entregan.RFC = Proveedores.RFC;  
SELECT RFC,Cantidad, Fecha,Numero  
FROM Proveedores_QueEmpiezancon_La  
WHERE Numero Between 5000 and 5010;
```

Consulta #2:

```
CREATE VIEW Top2_Proyectos  
AS SELECT * FROM Proyectos LIMIT 2;  
SELECT *  
FROM Top2_Proyectos;
```

Consulta #3:

```
CREATE VIEW Numero_Proyectos_Acaban_en_6  
AS SELECT Numero FROM Entregan WHERE Numero LIKE '___6';  
SELECT *  
FROM Numero_Proyectos_Acaban_en_6;
```

Consulta #4:

```
CREATE VIEW Entregas_1430_OR_1300  
AS SELECT * FROM entregan WHERE clave=1430 OR clave=1300;  
SELECT M.descripcion, M.clave, E.RFC, fecha, cantidad  
FROM Entregas_1430_OR_1300 AS E, Materiales AS M  
WHERE E.clave = M.clave;
```

Consulta #5:

```
CREATE VIEW Entregas_2000  
AS SELECT *  
FROM Entregan AS E  
WHERE fecha BETWEEN '2000-01-01' AND '2000-12-31';  
SELECT RFC, Avg(cantidad) AS 'Promedio de entregas'  
FROM Entregas_2000  
GROUP BY RFC;
```

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

```
SELECT M.clave, descripcion
FROM materiales AS M, Entregan AS E, proyectos AS P
WHERE M.clave = E.clave AND E.numero = P.numero AND
P.denominacion = 'México sin ti no estamos completos';
```

clave	descripcion

No existe ese proveedor.

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

```
SELECT M.clave, descripcion
FROM materiales AS M, Entregan AS E, Proveedores As Prov
WHERE M.clave = E.Clave AND E.rfc = Prov.rfc AND
Prov.razonsocial = 'Acme tools';
```

Tampoco existe Acme tools.

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

```
SELECT RFC, Avg(cantidad) AS 'Promedio de entregas'
FROM Entregan
WHERE fecha Between '2000-01-01' AND '2000-12-31'
GROUP BY RFC
HAVING Avg(cantidad) < 300;
```

RFC	'Promedio d

No sé si mi tabla esté desactualizada, pues las consultas son correctas pero no regresan datos.

El total entregado por cada material en el año 2000.

```
SELECT SUM(E.cantidad) AS 'Total entregado', M.descripcion
FROM Entregan AS E
WHERE E.fecha Between '2000-01-01' AND '2000-12-31'
GROUP BY E.clave;
```

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

```
SELECT SUM(cantidad), clave
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 denominacion, SUM((precio + impuesto) * cantidad) AS "Total a
pagar"
FROM Materiales AS M, Entregan AS E, Proyectos AS P
WHERE M.clave = E.clave AND E.numero = P.numero
GROUP BY denominacion
ORDER BY denominacion;
```

Denominación, RFC y RazonSocial de los proveedores que se suministran materiales al proyecto Televisa en acción que no se encuentran apoyando al proyecto Educando en Coahuila (Solo usando vistas).

```
CREATE VIEW Proyectos_Televisa AS
SELECT DISTINCT denominacion, Prov.RFC, Prov.RazonSocial
FROM proveedores AS Prov, entregan AS E, proyectos AS P
WHERE Prov.rfc = E.rfc AND E.numero = P.numero AND
denominacion = 'Televisa en acción';
CREATE VIEW Proyectos_Educando_Coahuila AS
SELECT Prov.RazonSocial
FROM proveedores AS Prov, entregan AS E, proyectos AS P
WHERE Prov.rfc = E.rfc AND E.numero = P.numero AND
P.denominacion = 'Educando en Coahuila';
SELECT DISTINCT denominacion, RFC, RazonSocial
FROM Proyectos_Televisa
WHERE RazonSocial NOT IN (SELECT RazonSocial FROM
Proyectos_Educando_Coahuila);
```

Denominación, RFC y RazonSocial de los proveedores que se suministran materiales al

proyecto Televisa en acción que no se encuentran apoyando al proyecto Educando en Coahuila (Sin usar vistas, utiliza not in, in o exists).

```
SELECT DISTINCT denominacion, Prov.RFC, Prov.RazonSocial
FROM proveedores AS Prov, entregan AS E, proyectos AS P
WHERE Prov.rfc = E.rfc AND E.numero = P.numero AND
denominacion = 'Televisa en acción' AND
Prov.RazonSocial NOT IN (SELECT Prov.RazonSocial
FROM proveedores AS Prov, entregan AS E, proyectos AS P
WHERE Prov.rfc = E.rfc AND E.numero = P.numero AND
P.denominacion = 'Educando en Coahuila');
```

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.

```
CREATE VIEW Materiales_Televisa AS
SELECT precio, descripcion, Prov.RazonSocial
FROM Materiales As M, entregan AS E, proyectos AS P, proveedores AS
Prov
WHERE M.clave = E.clave AND E.numero = P.numero AND E.rfc = Prov.rfc
AND
P.denominacion = 'Televisa en acción';
CREATE VIEW Proveedores_coahuila AS
SELECT Prov.RazonSocial
FROM Materiales As M, entregan AS E, proyectos AS P, proveedores AS
Prov
WHERE M.clave = E.clave AND E.numero = P.numero AND E.rfc = Prov.rfc
AND
P.denominacion = 'Educando en Coahuila';
SELECT precio, descripcion, RazonSocial
FROM Materiales_Televisa
WHERE RazonSocial IN (SELECT RazonSocial FROM Proveedores_coahuila);
```

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 'Cantidad Entregado',
Sum((M.precio + M.impuesto) * E.cantidad) AS 'Total de costo'
FROM Entregan As E, Materiales AS M
WHERE E.clave = M.clave
GROUP BY E.clave
ORDER BY M.descripcion;
```