

Tecnológico de Monterrey

Campus Querétaro

Construcción de software y toma de decisiones (Gpo 501)

Laboratorio 20:

“Consultas en SQL”

Carlos Alejandro Galicia Silva - A01709890

Profesores:

Enrique Alfonso Calderón Balderas

Denisse L. Maldonado Flores

Alejandro Fernández Vilchis

2 de Mayo de 2024



Consulta de una tabla completa:

The screenshot shows a SQL IDE interface. At the top, there are tabs for 'lab20.sql', 'entregan [@ahund]', 'materiales [@ahund]', and 'proyectos [@ahund]'. The 'lab20.sql' tab is active, and the query 'select * from materiales;' is entered in the editor. Below the editor, the 'Output' tab is selected, showing the results of the query. The results are displayed in a table with 45 rows. The table has four columns: 'clave', 'descripcion', 'precio', and 'impuesto'. The data is as follows:

clave	descripcion	precio	impuesto
1000	Varilla 3/16	100	10
1010	Varilla 4/32	115	11.5
1020	Varilla 3/17	130	13
1030	Varilla 4/33	145	14.5
1040	Varilla 3/18	160	16
1050	Varilla 4/34	175	17.5
1060	Varilla 3/19	190	19
1070	Varilla 4/35	205	20.5
1080	Ladrillos rojos	50	5
1090	Ladrillos grises	35	3.5
1100	Block	30	3
1110	Megablock	40	4
1120	Sillar rosa	100	10
1130	Sillar gris	110	11
1140	Cantera blanca	200	20
1150	Cantera gris	1210	121
1160	Cantera rosa	1420	142
1170	Cantera amarilla	230	23
1180	Recubrimiento P1001	200	20
1190	Recubrimiento P1010	220	22



Selección

5

6 ✓ `SELECT * FROM materiales WHERE clave = 1000;`

Output × lab20.materiales ×

1 row

	clave	descripcion	precio	impuesto
1	1000	Varilla 3/16	100	10

Proyección

7

8 ✓ `select clave,rfc,fecha from entregan;`

Output × lab20.entregan ×

87 rows

	clave	rfc	fecha
1	1000	AAAA800101	2001-12-13
2	1200	EEEE800101	2003-03-15
3	1400	AAAA800101	1999-04-07
4	1010	BBBB800101	1998-07-28
5	1210	FFFF800101	2000-05-21
6	1410	BBBB800101	2000-05-18
7	1020	CCCC800101	2003-12-16
8	1220	GGGG800101	2005-07-03
9	1420	CCCC800101	2001-09-09
10	1030	DDDD800101	1998-01-12
11	1230	HHHH800101	1998-09-12
12	1430	DDDD800101	2005-04-30
13	1040	EEEE800101	1999-12-18
14	1150	HHHH800101	2002-11-14
15	1240	AAAA800101	0000-00-00
16	1050	FFFF800101	2004-02-07
17	1140	GGGG800101	2000-06-30
18	1250	BBBB800101	2003-10-12
19	1300	GGGG800101	2004-02-28
20	1060	GGGG800101	2004-09-29



Reunión natural

```
10 select * from materiales,entregan
11 where materiales.clave = entregan.clave;
```

	materiales.clave	descripcion	precio	impuesto	entregan.clave	rfc	numero	fecha	cantidad
1	1000	Varilla 3/16	100	10	1000	AAAA800101	5000	2001-12-13	165
2	1000	Varilla 3/16	100	10	1000	AAAA800101	5019	1999-07-13	254
3	1010	Varilla 4/32	115	11.5	1010	BBBB800101	5001	1998-07-28	528
4	1010	Varilla 4/32	115	11.5	1010	BBBB800101	5018	1997-02-09	523
5	1020	Varilla 3/17	130	13	1020	CCCC800101	5002	2003-12-16	582
6	1020	Varilla 3/17	130	13	1020	CCCC800101	5017	2000-03-29	8
7	1030	Varilla 4/33	145	14.5	1030	DDDD800101	5003	1998-01-12	202
8	1030	Varilla 4/33	145	14.5	1030	DDDD800101	5016	2005-06-07	295
9	1040	Varilla 3/18	160	16	1040	EEEE800101	5004	1999-12-18	263
10	1040	Varilla 3/18	160	16	1040	EEEE800101	5015	1999-05-29	540
11	1050	Varilla 4/34	175	17.5	1050	FFFF800101	5005	2004-02-07	503
12	1050	Varilla 4/34	175	17.5	1050	FFFF800101	5014	2000-04-18	623
13	1060	Varilla 3/19	190	19	1060	GGGG800101	5006	2004-09-29	324
14	1060	Varilla 3/19	190	19	1060	GGGG800101	5013	2002-07-23	692
15	1070	Varilla 4/35	205	20.5	1070	HHHH800101	5007	2006-11-21	2
16	1070	Varilla 4/35	205	20.5	1070	HHHH800101	5012	2004-11-27	503
17	1080	Ladrillos rojos	50	5	1080	AAAA800101	5008	2005-04-03	86
18	1080	Ladrillos rojos	50	5	1080	AAAA800101	5011	2003-08-28	699

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

No debido a que se especifica en la consulta que la clave de la tabla material sea igual a la clave de la tabla entrega, por lo cual si existe un material con la clave que no este en la tabla entrega, no aparecería en el resultado de esta consulta.

Reunión con criterio específico

```
12
13 select * from entregan,proyectos
14 where entregan.numero <= proyectos.numero;
```

	clave	rfc	entregan.numero	fecha	cantidad	proyectos.numero	denominacion
1	1000	AAAA800101	5000	2001-12-13	165	5000	Vamos Mexico
2	1200	EEEE800101	5000	2003-03-15	177	5000	Vamos Mexico
3	1400	AAAA800101	5000	1999-04-07	382	5000	Vamos Mexico
4	1000	AAAA800101	5000	2001-12-13	165	5001	Aztecon
5	1200	EEEE800101	5000	2003-03-15	177	5001	Aztecon
6	1400	AAAA800101	5000	1999-04-07	382	5001	Aztecon
7	1010	BBBB800101	5001	1998-07-28	528	5001	Aztecon
8	1210	FFFF800101	5001	2000-05-21	43	5001	Aztecon
9	1410	BBBB800101	5001	2000-05-18	601	5001	Aztecon
10	1000	AAAA800101	5000	2001-12-13	165	5002	CIT Campeche
11	1200	EEEE800101	5000	2003-03-15	177	5002	CIT Campeche
12	1400	AAAA800101	5000	1999-04-07	382	5002	CIT Campeche
13	1010	BBBB800101	5001	1998-07-28	528	5002	CIT Campeche
14	1210	FFFF800101	5001	2000-05-21	43	5002	CIT Campeche
15	1410	BBBB800101	5001	2000-05-18	601	5002	CIT Campeche
16	1020	CCCC800101	5002	2003-12-16	582	5002	CIT Campeche
17	1220	GGGG800101	5002	2005-07-03	24	5002	CIT Campeche
18	1420	CCCC800101	5002	2001-09-09	603	5002	CIT Campeche



Unión (se ilustra junto con selección)

```
16 ✓ (select * from entregan where clave=1450)
17 union
18 (select * from entregan where clave=1300);
```

as_cargar

Output × lab20.entregan ×

2 rows

	clave	rfc	numero	fecha	cantidad
1	1300	GGGG800101	5005	2004-02-28	521
2	1300	GGGG800101	5010	2001-02-10	119

¿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)

```
21
22 ✓ select clave from entregan where numero=5001
23 AND clave NOT IN
24 (select clave from entregan where numero=5018);
25
```

argar

Output × lab20.entregan ×

2 rows

	clave
1	1210
2	1410



Diferencia (se ilustra con selección)

```
25
26 ✓ select * from entregan e1
27 WHERE NOT EXISTS
28 (select 1 from entregan e2 where e2.clave=1000 AND e1.clave = e2.clave);
```

Output × lab20.entregan ×

85 rows

	clave	rfc	numero	fecha	cantidad
1	1010	BBBB800101	5001	1998-07-28	528
2	1010	BBBB800101	5018	1997-02-09	523
3	1020	CCCC800101	5002	2003-12-16	582
4	1020	CCCC800101	5017	2000-03-29	8
5	1030	DDDD800101	5003	1998-01-12	202
6	1030	DDDD800101	5016	2005-06-07	295
7	1040	EEEE800101	5004	1999-12-18	263
8	1040	EEEE800101	5015	1999-05-29	540
9	1050	FFFF800101	5005	2004-02-07	503
10	1050	FFFF800101	5014	2000-04-18	623
11	1060	GGGG800101	5006	2004-09-29	324
12	1060	GGGG800101	5013	2002-07-23	692
13	1070	HHHH800101	5007	2006-11-21	2
14	1070	HHHH800101	5012	2004-11-27	503
15	1080	AAAA800101	5008	2005-04-03	86
16	1080	AAAA800101	5011	2003-08-28	699
17	1090	BBBB800101	5009	1997-03-13	73
18	1090	BBBB800101	5010	1998-11-17	421



Producto cartesiano

	entregan.clave	rfc	numero	fecha	cantidad	materiales.clave	descripcion	precio	impuesto
1	1000	AAAA800101	5000	2001-12-13	165	1000	Varilla 3/16	100	10
2	1000	AAAA800101	5000	2001-12-13	165	1010	Varilla 4/32	115	11.5
3	1000	AAAA800101	5000	2001-12-13	165	1020	Varilla 3/17	130	13
4	1000	AAAA800101	5000	2001-12-13	165	1030	Varilla 4/33	145	14.5
5	1000	AAAA800101	5000	2001-12-13	165	1040	Varilla 3/18	160	16
6	1000	AAAA800101	5000	2001-12-13	165	1050	Varilla 4/34	175	17.5
7	1000	AAAA800101	5000	2001-12-13	165	1060	Varilla 3/19	190	19
8	1000	AAAA800101	5000	2001-12-13	165	1070	Varilla 4/35	205	20.5
9	1000	AAAA800101	5000	2001-12-13	165	1080	Ladrillos rojos	50	5
10	1000	AAAA800101	5000	2001-12-13	165	1090	Ladrillos grises	35	3.5
11	1000	AAAA800101	5000	2001-12-13	165	1100	Block	30	3
12	1000	AAAA800101	5000	2001-12-13	165	1110	Megablock	40	4
13	1000	AAAA800101	5000	2001-12-13	165	1120	Sillar rosa	100	10
14	1000	AAAA800101	5000	2001-12-13	165	1130	Sillar gris	110	11
15	1000	AAAA800101	5000	2001-12-13	165	1140	Cantera blanca	200	20
16	1000	AAAA800101	5000	2001-12-13	165	1150	Cantera gris	1210	121
17	1000	AAAA800101	5000	2001-12-13	165	1160	Cantera rosa	1420	142
18	1000	AAAA800101	5000	2001-12-13	165	1170	Cantera amarilla	230	23
19	1000	AAAA800101	5000	2001-12-13	165	1180	Recubrimiento P1001	200	20
20	1000	AAAA800101	5000	2001-12-13	165	1190	Recubrimiento P1010	220	22

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

El numero de tablas en el resultado de la consulta es igual a la multiplicación del numero de tuplas de la tabla Entregan(87) multiplicado por el numero de tuplas de la tabla Materiales(45), lo que es equivalente a 3915 tuplas.

Construcción de consultas a partir de una especificación

```
32 select descripcion FROM materiales m
33 JOIN entregan e on m.clave = e.clave
34 WHERE YEAR(e.fecha) = 2000;
```

	descripcion
1	Varilla 3/17
2	Varilla 4/34
3	Block
4	Sillar gris
5	Sillar gris
6	Cantera blanca
7	Recubrimiento P1028
8	Tubería 3.6
9	Pintura C1010
10	Pintura B1021
11	Pintura B1021
12	Pintura B1022



¿Por qué aparecen varias veces algunas descripciones de material?
Debido a que el mismo material se entregó en diferentes fechas que pueden pertenecer al mismo año 2000

Uso del calificador distinct

```
32 ✓ SELECT DISTINCT descripcion FROM materiales m
33     JOIN entregan e on m.clave = e.clave
34     WHERE YEAR(e.fecha) = 2000;
```

Output × lab20.materiales ×

10 rows

	descripcion
1	Varilla 3/17
2	Varilla 4/34
3	Block
4	Sillar gris
5	Cantera blanca
6	Recubrimiento P1028
7	Tubería 3.6
8	Pintura C1010
9	Pintura B1021
10	Pintura B1022

¿Qué resultado obtienes en esta ocasión?
Los registros de las descripciones de los materiales entregados en el año 2000 pero sin repetirse, solo registrando una fecha de una instancia del material en específico.



Ordenamientos

```
36 ✓ SELECT p.numero, p.denominacion, e.fecha, e.cantidad
37 FROM proyectos p
38 JOIN entregan e ON p.numero = e.numero
39 ORDER BY p.numero, e.fecha DESC;
```

Output × Result 37 ×

|< < 87 rows > >| ↺ ⌵ ⚡

	numero	denominacion	fecha	cantidad
1	5000	Vamos Mexico	2003-03-15	177
2	5000	Vamos Mexico	2001-12-13	165
3	5000	Vamos Mexico	1999-04-07	382
4	5001	Aztecon	2000-05-21	43
5	5001	Aztecon	2000-05-18	601
6	5001	Aztecon	1998-07-28	528
7	5002	CIT Campeche	2005-07-03	24
8	5002	CIT Campeche	2003-12-16	582
9	5002	CIT Campeche	2001-09-09	603
10	5003	Mexico sin ti no estamos completos	2005-04-30	576
11	5003	Mexico sin ti no estamos completos	1998-09-12	530
12	5003	Mexico sin ti no estamos completos	1998-01-12	202
13	5004	Educando en Coahuila	2002-11-14	453
14	5004	Educando en Coahuila	1999-12-18	263
15	5004	Educando en Coahuila	0000-00-00	152
16	5005	Infonavit Durango	2004-02-28	521
17	5005	Infonavit Durango	2004-02-07	503
18	5005	Infonavit Durango	2003-10-12	71



Uso de expresiones

```
41 ✓ SELECT clave, descripcion, (precio * impuesto) as precio_total
42 FROM materiales;
```

Output × Result 44 ×

45 rows

	clave	descripcion	precio_total
1	1000	Varilla 3/16	1000
2	1010	Varilla 4/32	1322.5
3	1020	Varilla 3/17	1690
4	1030	Varilla 4/33	2102.5
5	1040	Varilla 3/18	2560
6	1050	Varilla 4/34	3062.5
7	1060	Varilla 3/19	3610
8	1070	Varilla 4/35	4202.5
9	1080	Ladrillos rojos	250
10	1090	Ladrillos grises	122.5
11	1100	Block	90
12	1110	Megablock	160
13	1120	Sillar rosa	1000
14	1130	Sillar gris	1210
15	1140	Cantera blanca	4000
16	1150	Cantera gris	146410
17	1160	Cantera rosa	201640
18	1170	Cantera amarilla	5290

Operadores de cadena

```
44 ✓ SELECT * FROM materiales where Descripcion LIKE 'Si%';
```

Output × lab20.materiales ×

2 rows

	clave	descripcion	precio	impuesto
1	1120	Sillar rosa	100	10
2	1130	Sillar gris	110	11

¿Qué resultado obtienes? Los registros de los materiales cuya descripcion empieza por "SI"

Explica que hace el símbolo '%'. Representa al resto de caracteres del string



¿Qué sucede si la consulta fuera : LIKE 'SI' ? Devolvería los registros de los materiales cuya descripción es igual a "SI"

¿Qué resultado obtienes? No devuelve registros

Explica a qué se debe este comportamiento. Debido a que la condición de la consulta no tiene el "%" ya no busca el resto del string empezando por "SI", sino que busca el resultado solo si es igual a los caracteres especificados

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

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

Muestra el RFC de las entregas cuyo RFC empiece con la A, B, C o D, utilizando el signo "-" para establecer un rango y el "%" para representar al resto de caracteres

The screenshot shows a SQL IDE interface. At the top, a query is entered in a text area: `SELECT RFC FROM Entregan WHERE RFC LIKE '[A-D]%' ;`. Below the query area, there is a tab labeled "lab20.entregan". The results are displayed in a table with 47 rows. The first 11 rows are visible, showing RFC values that start with A, B, C, or D. The table has a column header "RFC" and a list of 11 rows of data.

	RFC
1	AAAA800101
2	AAAA800101
3	BBBB800101
4	BBBB800101
5	CCCC800101
6	CCCC800101
7	DDDD800101
8	DDDD800101
9	AAAA800101
10	BBBB800101
11	CCCC800101



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

Muestra el RFC de las entregas cuyo RFC no empiece con la letra A, utilizando el signo "^" para que coincida con cualquier caracter menos el especificado.

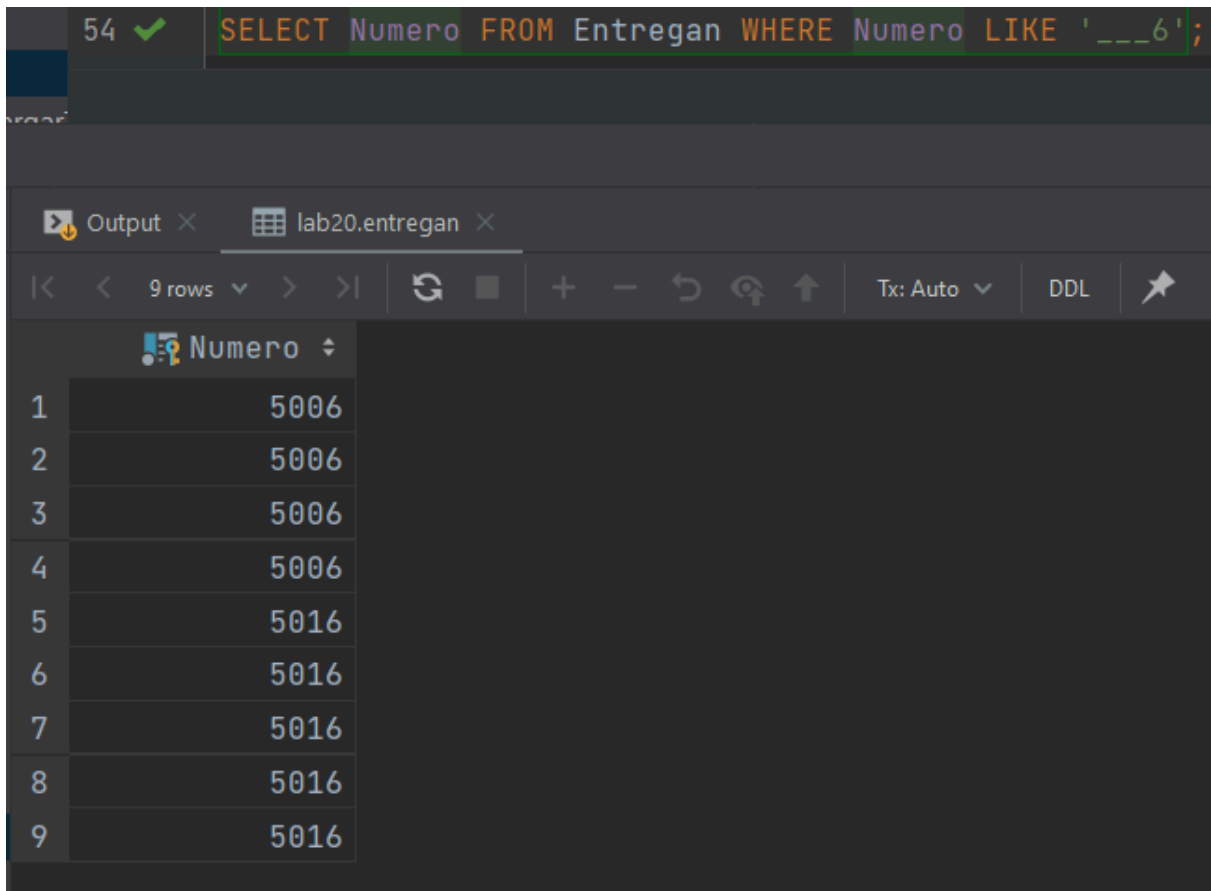
53	SELECT RFC FROM Entregan WHERE RFC LIKE '[^A]%';
----	--

Output x lab20.entregan x	
75 rows	
	RFC
1	EEEE800101
2	BBBB800101
3	FFFF800101
4	BBBB800101
5	CCCC800101
6	GGGG800101
7	CCCC800101
8	DDDD800101
9	HHHH800101
10	DDDD800101
11	EEEE800101
12	HHHH800101
13	FFFF800101
14	GGGG800101
15	BBBB800101
16	GGGG800101
17	GGGG800101
18	FFFF800101
19	CCCC800101



SELECT Numero FROM Entregan WHERE Numero LIKE '___6';

Muestra el numero de las entregas donde el ultimo digito del numero sea igual a 6 utilizando el signo "_" para coincidir con cada digito o caracter en la cadena.



54 ✓ `SELECT Numero FROM Entregan WHERE Numero LIKE '___6';`

Output × lab20.entregan ×

9 rows

	Numero
1	5006
2	5006
3	5006
4	5006
5	5016
6	5016
7	5016
8	5016
9	5016



Operadores Lógicos

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

Output x lab20.entregan x

43 rows

	Clave	RFC	Numero	Fecha	Cantidad
1	1000	AAAA800101	5000	2001-12-13	165
2	1010	BBBB800101	5001	1998-07-28	528
3	1020	CCCC800101	5002	2003-12-16	582
4	1030	DDDD800101	5003	1998-01-12	202
5	1040	EEEE800101	5004	1999-12-18	263
6	1050	FFFF800101	5005	2004-02-07	503
7	1060	GGGG800101	5006	2004-09-29	324
8	1070	HHHH800101	5007	2006-11-21	2
9	1080	AAAA800101	5008	2005-04-03	86
10	1090	BBBB800101	5009	1997-03-13	73
11	1090	BBBB800101	5010	1998-11-17	421
12	1100	CCCC800101	5009	2000-12-07	466
13	1100	CCCC800101	5010	2001-11-19	699
14	1110	DDDD800101	5008	2005-06-03	337
15	1120	EEEE800101	5007	2003-11-21	692
16	1130	FFFF800101	5006	2000-04-13	562
17	1140	GGGG800101	5005	2000-06-30	583
18	1150	HHHH800101	5004	2002-11-14	453
19	1200	FFFF800101	5000	2003-03-15	177

¿Cómo filtrarías rangos de fechas?

Utilizando YEAR(fecha) BETWEEN 2000 and 2010 en la consulta



```
60 ✓ SELECT RFC,Cantidad, Fecha,Numero
61 FROM Entregan
62 WHERE Numero Between 5000 and 5010 AND
63 Exists ( SELECT RFC
64 FROM Proveedores
65 WHERE RazonSocial LIKE 'La%' and Entregan.RFC = Proveedores.RFC);
```

Output x lab20.entregan x

12 rows

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165	2001-12-13	5000
2	AAAA800101	86	2005-04-03	5008
3	AAAA800101	152	0000-00-00	5004
4	AAAA800101	382	1999-04-07	5000
5	AAAA800101	116	2005-04-21	5010
6	CCCC800101	582	2003-12-16	5002
7	CCCC800101	466	2000-12-07	5009
8	CCCC800101	699	2001-11-19	5010
9	CCCC800101	460	2001-04-09	5006
10	CCCC800101	631	2001-07-28	5009
11	CCCC800101	603	2001-09-09	5002
12	CCCC800101	278	1999-05-05	5008

¿Qué hace la consulta? La consulta selecciona entregas (RFC, Cantidad, Fecha, Numero) que cumplen con dos condiciones: el número de entrega está entre 5000 y 5010, y el proveedor asociado con la entrega tiene una razón social que comienza con 'La'.

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

Indica una subconsulta que busca proveedores cuya razón social comience con 'La' y cuyo RFC coincida con el RFC de la tabla Entregan.



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

```
69 WHERE Numero BETWEEN 5000 AND 5010
70 AND RFC IN (
71     SELECT RFC
72     FROM Proveedores
73     WHERE RazonSocial LIKE 'La%'
74 );
```

Output x lab20.entregan x

12 rows

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165	2001-12-13	5000
2	AAAA800101	86	2005-04-03	5008
3	AAAA800101	152	0000-00-00	5004
4	AAAA800101	382	1999-04-07	5000
5	AAAA800101	116	2005-04-21	5010
6	CCCC800101	582	2003-12-16	5002
7	CCCC800101	466	2000-12-07	5009
8	CCCC800101	699	2001-11-19	5010
9	CCCC800101	460	2001-04-09	5006
10	CCCC800101	631	2001-07-28	5009
11	CCCC800101	603	2001-09-09	5002
12	CCCC800101	278	1999-05-05	5008



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.

```
77 FROM Entregan
78 WHERE Numero BETWEEN 5000 AND 5010
79 AND RFC NOT IN (
80     SELECT RFC
81     FROM Proveedores
82     WHERE RazonSocial NOT LIKE 'La%'
83 );
```

Output x lab20.entregan x

12 rows

	RFC	Cantidad	Fecha	Numero
1	AAAA800101	165	2001-12-13	5000
2	CCCC800101	582	2003-12-16	5002
3	AAAA800101	86	2005-04-03	5008
4	CCCC800101	466	2000-12-07	5009
5	CCCC800101	699	2001-11-19	5010
6	AAAA800101	152	0000-00-00	5004
7	CCCC800101	460	2001-04-09	5006
8	CCCC800101	631	2001-07-28	5009
9	AAAA800101	382	1999-04-07	5000
10	AAAA800101	116	2005-04-21	5010
11	CCCC800101	603	2001-09-09	5002
12	CCCC800101	278	1999-05-05	5008



¿Qué hace la siguiente sentencia? Explica por qué.

`SELECT TOP 2 * FROM Proyectos`

Selecciona las dos primeras filas de la tabla Proyectos y devuelve todas las columnas de esas filas debido a que utiliza el operador TOP con un valor de 2

85 ✓
86

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

Output × lab20.proyectos × lab20.proyectos 2

2 rows

	numero	denominacion
1	5000	Vamos Mexico
2	5001	Aztecon

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

`SELECT TOP Numero FROM Proyectos`

Selecciona la primera fila de la tabla Proyectos y devuelve todas las columnas de esa fila debido a que utiliza el operador TOP con un valor default de 1

87 ✓
88

```
SELECT * FROM Proyectos LIMIT 1;
```

Output × lab20.proyectos × lab20.proyectos 2 ×

1 row

	numero	denominacion
1	5000	Vamos Mexico



Modificando la estructura de un tabla existente.

¿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.clave, e.cantidad, m.precio, m.impuesto, (m.precio * (1 + (m.impuesto / 100)))
as precioTotal
FROM entregan e
JOIN materiales m ON e.clave = m.clave;
```

Creación de vistas

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

```
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';
```

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

```
SELECT m.Clave, m.Descripcion
FROM Materiales m
JOIN entregan e on m.clave = e.clave
JOIN Proveedores p ON p.RFC = e.RFC
WHERE p.RazonSocial = 'Acme tools';
```

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

```
SELECT RFC
FROM Entregan
WHERE YEAR(Fecha) = 2000
GROUP BY RFC
HAVING AVG(Cantidad) >= 300;
```

El Total entregado por cada material en el año 2000.

```
SELECT clave, SUM(cantidad) AS TotalEntregado
FROM entregan
WHERE YEAR(Fecha) = 2000
GROUP BY clave;
```



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

```
CREATE VIEW MaterialMasVendido AS
SELECT clave, SUM(Cantidad) AS TotalVentas
FROM Entregan
WHERE YEAR(Fecha) = 2001
GROUP BY clave;
```

```
SELECT clave
FROM MaterialMasVendido
ORDER BY TotalVentas DESC LIMIT 1;
```

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

```
SELECT *
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) AS TotalPagar
FROM entregan e
JOIN Materiales m ON e.clave = m.Clave
JOIN Proyectos p ON e.numero = p.numero
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 Proveedores_Televisa AS
SELECT DISTINCT p.rfc, p.razonsocial, pr.denominacion
FROM proveedores p
JOIN entregan e ON p.rfc = e.rfc
JOIN materiales m ON m.clave = e.clave
JOIN proyectos pr ON e.numero = pr.numero
WHERE pr.denominacion = 'Televisa en acción';
```

```
CREATE VIEW Proveedores_Coahuila AS
SELECT DISTINCT p.rfc, p.razonsocial, pr.denominacion
FROM proveedores p
JOIN entregan e ON p.rfc = e.rfc
JOIN materiales m ON m.clave = e.clave
JOIN proyectos pr ON e.numero = pr.numero
WHERE pr.denominacion = 'Educando en Coahuila';
```



```
SELECT p.denominacion, p.rfc, p.razonsocial
FROM Proveedores_Televisa p
LEFT JOIN Proveedores_Coahuila pc ON p.rfc = pc.rfc
WHERE pc.rfc IS NULL;
```

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 pr.denominacion, p.rfc, p.razonsocial
FROM proveedores p
JOIN entregan e ON p.rfc = e.rfc
JOIN materiales m ON m.clave = e.clave
JOIN proyectos pr ON e.numero = pr.numero
WHERE pr.denominacion = 'Televisa en acción'
AND p.rfc NOT IN (
    SELECT p.rfc
    FROM proveedores p
    JOIN entregan e ON p.rfc = e.rfc
    JOIN materiales m ON m.clave = e.clave
    JOIN proyectos pr ON e.numero = pr.numero
    WHERE pr.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.

```
SELECT m.descripcion, m.precio, e.cantidad
FROM materiales m
JOIN entregan e ON m.clave = e.clave
WHERE e.numero = 5008 -- Proyecto Televisa en acción
AND EXISTS (
    SELECT 1
    FROM entregan e2
    JOIN proyectos pr ON e2.numero = pr.numero
    JOIN materiales m2 ON e2.clave = m2.clave
    WHERE pr.denominacion = 'Educando en Coahuila'
    AND e.rfc = e2.rfc
);
```



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

```
SELECT m.descripcion, COUNT(*), SUM(m.precio * e.cantidad)
FROM materiales m
JOIN entregan e ON m.clave = e.clave
GROUP BY m.clave, m.descripcion;
```