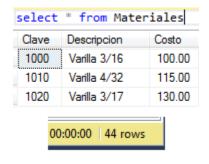
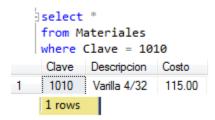
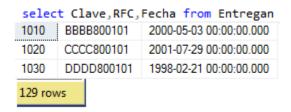
Consulta de una tabla completa



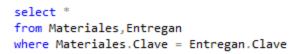
Selección



Proyección



Reunión Natural

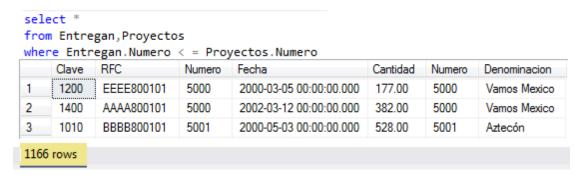


	Clave	Descripcion	Costo	Clave	RFC	Numero	Fecha	Cantidad
1	1010	Varilla 4/32	115.00	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00
2	1010	Varilla 4/32	115.00	1010	BBBB800101	5018	2002-03-29 00:00:00.000	523.00
3	1010	Varilla 4/32	115.00	1010	BBBB800101	5018	2000-11-10 00:00:00.000	667.00
129 rows								

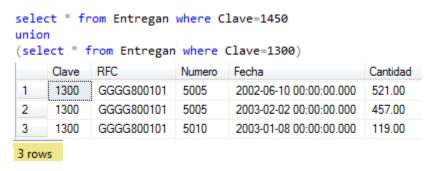
Si algún material no ha se ha entregado ¿Aparecería en el resultado de esta consulta?

No, porque se especifica que las claves de las tablas deben de coincidir.

Reunión con criterio específico



Unión (se ilustra junto con selección)



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

Select * from Entregan where Clave = 1450 or Clave = 1300

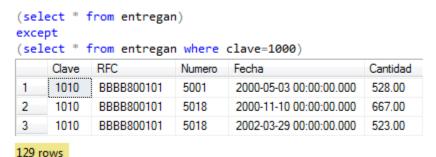
Intersección (se ilustra junto con selección y proyección)

```
select Clave from entregan where numero=5001
intersect
(select Clave from entregan where numero=5018)

Clave
1 1010

1 rows
```

Diferencia (se ilustra con selección)



Producto cartesiano

select *	from	entregan, materiales
----------	------	----------------------

	Clave	RFC	Numero	Fecha	Cantidad	Clave	Descripcion	Costo
1	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00	1000	Varilla 3/16	100.00
2	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00	1000	Varilla 3/16	100.00
3	1020	CCCC800101	5002	2001-07-29 00:00:00.000	582.00	1000	Varilla 3/16	100.00
5808	5808 rows							

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

Hace un producto cartesiano de las tuplas en ambas tablas, el número de filas es igual a la multiplicación del número de filas en entregan y el número de filas en materiales.

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 no se está usando distinct para evitar repeticiones.

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?

Los resultados sin repeticiones entre las tuplas.

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.

```
select p.Numero, p.Denominacion, e.Fecha, e.Cantidad
from Proyectos p, Entregan e
where p.Numero = e.Numero
group by p.Numero, p.Denominacion, e.Fecha, e.Cantidad
order by e.Fecha desc
```

	Numero	Denominacion	Fecha	Cantidad
1	5004	Educando en Coahuila	2003-09-01 00:00:00.000	270.00
2	5011	Remodelación de aulas del IPP	2003-08-05 00:00:00.000	549.00
3	5011	Remodelación de aulas del IPP	2003-06-28 00:00:00.000	368.00
132	rows			

Operadores de cadena

SELECT * FROM productos where Descripcion LIKE 'Si%'

SELECT * FROM Proyectos where Denominacion

Numero Denominacion

Source Reconstrucción del templo de Guadalupe

Source Remodelación de aulas del IPP

Source Restauración de instalaciones del CEA

¿Qué resultado obtienes?

Todos los proyectos con "re" al principio de su denominación

Explica que hace el símbolo '%'.

Es un comodín para ignorar el texto, excepto lo del patrón que gueramos.

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

Va a buscar que exactamente sea 'Si'.

¿Qué resultado obtienes?

Ninguno.

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?

¿Qué resultado obtienes? ¿¿¿??? ¿Para qué sirve DECLARE?

Para declarar una variable. ¿Cuál es la función de @foo?

Almacenar una cadena. ¿Que realiza el operador SET?

Para asignar 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 '[A-D]%'; SELECT RFC FROM Entregan WHERE RFC LIKE '[A]%'; SELECT Numero FROM Entregan WHERE Numero LIKE '___6'; ¿Cómo filtrarías rangos de fechas?
```

Con una consulta tipo 'fecha' between 'fecha1' and 'fecha2'

EXISTS

```
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])
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
    RFC
                Cantidad Fecha
                                           Numero
     AAAA800101 165.00 1998-07-08 00:00:00.000
                                           5000
                        2001-07-29 00:00:00.000
     CCCC800101 582.00
                                           5002
3 AAAA800101 86.00
                       1999-01-12 00:00:00.000 5008
16 rows
```

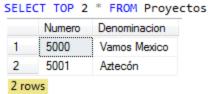
¿Qué hace la consulta?

Despliega el RFC, la cantidad de material entregada por el proveedor, la fecha en la que la entrego y el número de proyecto al que se realizó la entrega. ¿Qué función tiene el paréntesis () después de EXISTS?

Es para especificar qué es lo que existe.

¿Qué hace la siguiente sentencia? Explica por qué.

SELECT TOP 2 * FROM Proyectos



Despliega las 2 primeras tuplas de la tabla proyectos ¿Qué sucede con la siguiente consulta? Explica por qué.

SELECT TOP Numero FROM Proyectos

No desplego nada, en su lugar marco un error, tal parece que se necesita un valor constante para decir que top es el que va a tomar.

Modificando la estructura de una 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?

Creación de vistas

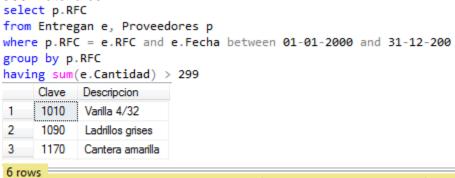
Los materiales (clave y descripción) entregados al proyecto "México sin ti no estamos completos".

```
select m.Clave, m.Descripcion
from Materiales m, Entregan e, Proyectos p
where m.Clave = e.Clave and p.Numero = e.Numero
      and p.Denominacion = 'Mexico sin ti no estamos completos'
group by m.Clave, m.Descripcion
     Clave
            Descripcion
     1030
            Varilla 4/33
 1
 2
      1230 Cemento
      1430 Pintura B1022
 3
3 rows
                                 10.25.18.106,10433 (10.0 SP3) a1703446 (
```

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 m.Clave = e.Clave and p.RFC = e.RFC | and p.RazonSocial = 'Oviedo' | group by m.Clave, m.Descripcion | Clave | Descripcion | 1 | 1010 | Varilla 4/32 | 2 | 1090 | Ladrillos grises | 3 | 1170 | Cantera amarilla | 6 rows | 10.25.18.106
```

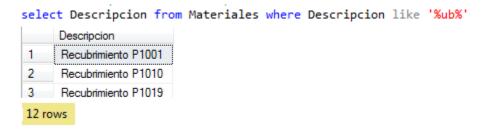
El RFC de los proveedores que durante el 2000 entregaron en promedio cuando menos 300 materiales.



El Total entregado por cada material en el año 2000.

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

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



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

```
select p.Denominacion, sum((m.Costo + m.PorcentajeImpuesto) * e.Cantidad) as 'Importe total'
from Entregan e, Materiales m, Proyectos p
where e.Clave = m.Clave and p.Numero = e.Numero
group by p.Denominacion

Denominacion | Importe total

Ampliación de la carretera a la huasteca | 731303.4000

Aztecón | 149460.4400

CIT Campeche | 160713.3600
```

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 proveedoresEducandoCoahuila as
select p.Denominacion, pr.RFC, pr.RazonSocial
from Proyectos p, Proveedores pr, Entregan e
where p.Numero = e.Numero and pr.RFC = e.RFC
            and p.Denominacion = 'Educando en Coahuila'
group by p.Denominacion, pr.RFC, pr.RazonSocial
create view proveedoresTelevisa as
select p.Denominacion, pr.RFC, pr.RazonSocial
from Proyectos p, Proveedores pr, Entregan e
where p.Numero = e.Numero and pr.RFC = e.RFC
            and p.Denominacion = 'Televisa en acción'
group by p.Denominacion, pr.RFC, pr.RazonSocial
create view tnc as
select RFC from proveedoresTelevisa
select RFC from proveedoresEducandoCoahuila
select pt.Denominacion, pt.RFC, pt.RazonSocial
from proveedoresTelevisa pt, tnc
where pt.RFC = tnc.RFC
group by pt.Denominacion, pt.RFC, pt.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).

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 CoahuilaYTelevisa as
select RFC from proveedoresEducandoCoahuila
intersect
select RFC from proveedoresTelevisa

select pt.Denominacion, m.Descripcion, sum(m.Costo + m.PorcentajeImpuesto) as 'Costo total'
from Materiales m, Entregan e, CoahuilaYTelevisa cyt, proveedoresTelevisa pt
where m.Clave = e.Clave and cyt.RFC = e.RFC
group by m.Descripcion, pt.Denominacion
```