

Consulta de una tabla completa

```
select * from Materiales
```

Clave	Descripcion	Costo
1000	Vanilla 3/16	100.00
1010	Vanilla 4/32	115.00
1020	Vanilla 3/17	130.00

00:00:00 | 44 rows

Selección

```
select *  
from Materiales  
where Clave = 1010
```

	Clave	Descripcion	Costo
1	1010	Vanilla 4/32	115.00

1 rows

Proyección

```
select Clave,RFC,Fecha from Entregan
```

1010	BBBB800101	2000-05-03 00:00:00.000
1020	CCCC800101	2001-07-29 00:00:00.000
1030	DDDD800101	1998-02-21 00:00:00.000

129 rows

Reunión Natural

```
select *  
from Materiales,Entregan  
where Materiales.Clave = Entregan.Clave
```

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

```
select *  
from Entregan,Proyectos  
where Entregan.Numero < = Proyectos.Numero
```

	Clave	RFC	Numero	Fecha	Cantidad	Numero	Denominacion
1	1200	EEEE800101	5000	2000-03-05 00:00:00.000	177.00	5000	Vamos Mexico
2	1400	AAAA800101	5000	2002-03-12 00:00:00.000	382.00	5000	Vamos Mexico
3	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00	5001	Aztecón

1166 rows

Unión (se ilustra junto con selección)

```
select * from Entregan where Clave=1450  
union  
(select * from Entregan where Clave=1300)
```

	Clave	RFC	Numero	Fecha	Cantidad
1	1300	GGGG800101	5005	2002-06-10 00:00:00.000	521.00
2	1300	GGGG800101	5005	2003-02-02 00:00:00.000	457.00
3	1300	GGGG800101	5010	2003-01-08 00:00:00.000	119.00

3 rows

¿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)

```
(select * from entregan)  
except  
(select * from entregan where clave=1000)
```

	Clave	RFC	Numero	Fecha	Cantidad
1	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00
2	1010	BBBB800101	5018	2000-11-10 00:00:00.000	667.00
3	1010	BBBB800101	5018	2002-03-29 00:00:00.000	523.00

129 rows

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	Vanilla 3/16	100.00
2	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00	1000	Vanilla 3/16	100.00
3	1020	CCCC800101	5002	2001-07-29 00:00:00.000	582.00	1000	Vanilla 3/16	100.00

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 LIKE 're%'
```

	Numero	Denominacion
1	5006	Reconstrucción del templo de Guadalupe
2	5011	Remodelación de aulas del IPP
3	5012	Restauración de instalaciones del CEA

6 rows

¿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 queramos.

¿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
1	AAAA800101	165.00	1998-07-08 00:00:00.000	5000
2	CCCC800101	582.00	2001-07-29 00:00:00.000	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
```

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

	Numero	Denominacion
1	5000	Vamos Mexico
2	5001	Aztecón

2 rows

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

SELECT TOP Numero FROM Proyectos

No despliego 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?

```
select e.Clave, sum((m.Costo + m.PorcentajeImpuesto) * e.Cantidad) as 'Importe total'
from Entregan e, Materiales m
where e.Clave = m.Clave
group by e.Clave
order by [Importe total] desc
```

	Clave	Importe total
1	1160	1523304.7200
2	1150	1431726.3000
3	1380	796169.4400

44 rows

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
1	1030	Vanilla 4/33
2	1230	Cemento
3	1430	Pintura B1022

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.

```
select p.RFC
from Entregan e, Proveedores p
where p.RFC = e.RFC and e.Fecha between 01-01-2000 and 31-12-2000
group by p.RFC
having sum(e.Cantidad) > 299
```

	Clave	Descripcion
1	1010	Varilla 4/32
2	1090	Ladrillos grises
3	1170	Cantera amarilla

6 rows | 10.25.18.106

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.

```
select Descripcion from Materiales where Descripcion like '%ub%'
```

	Descripcion
1	Recubrimiento P1001
2	Recubrimiento P1010
3	Recubrimiento P1019

12 rows

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
1	Ampliación de la carretera a la huasteca	731303.4000
2	Aztecón	149460.4400
3	CIT Campeche	160713.3600

20 rows

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
except
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).


```
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'
    and pr.RFC not in
    (
        select pr.RFC
        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
```

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
```