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.

**SQL  
select \* from materiales**

1000 "Varilla 3/16" 100.00

1010 "Varilla 4/32" 115.00

1020 "Varilla 3/17" 130.00

Total 44.

**select \* from materiales  
where clave=1000**

1000 "Varilla 3/16" 100.00

Total 1

**SQL**

**select clave,rfc,fecha from entregan**

1000 "AAAA800101" "1998-07-08 00:00:00.000"

1000 "AAAA800101" "1999-08-08 00:00:00.000"

1000 "AAAA800101" "2000-04-06 00:00:00.000"

Total: 132

**SQL**

**select \* from materiales,entregan  
where materiales.clave = entregan.clave**

1000 "Varilla 3/16" 100.00 1000 "AAAA800101" 5000 "1998-07-08 00:00:00.000" 165.00

1000 "Varilla 3/16" 100.00 1000 "AAAA800101" 5019 "1999-08-08 00:00:00.000" 254.00

1000 "Varilla 3/16" 100.00 1000 "AAAA800101" 5019 "2000-04-06 00:00:00.000" 7.00

Total: 132

Si algún material no ha se ha entregado ¿Aparecería en el resultado de esta consulta? NO, porque no encontraría el ID en la tabla.

**SQL**

**select \* from entregan,proyectos  
where entregan.numero < = proyectos.numero**

1000 "AAAA800101" 5000 "1998-07-08 00:00:00.000" 165.00 5000 "Vamos Mexico"

1200 "EEEE800101" 5000 "2000-03-05 00:00:00.000" 177.00 5000 "Vamos Mexico"

1400 "AAAA800101" 5000 "2002-03-12 00:00:00.000" 382.00 5000 "Vamos Mexico"

Total: 1188

**SQL  
(select \* from entregan where clave=1450)  
union  
(select \* from entregan where clave=1300)**

1300 "GGGG800101" 5005 "2002-06-10 00:00:00.000" 521.00

1300 "GGGG800101" 5005 "2003-02-02 00:00:00.000" 457.00

1300 "GGGG800101" 5010 "2003-01-08 00:00:00.000" 119.00

Total: 3

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

**SELECT \***

**FROM Entregan**

**Where clave = 1450 OR clave = 1300**

**SQL**

**(select clave from entregan where numero=5001)  
intersect  
(select clave from entregan where numero=5018)**

1010

Total: 1

SQL  
(select \* from entregan)  
minus  
(select \* from entregan where clave=1000)

Nuevamente, "minus" es una palabra reservada que no está definida en SQL Server, define una consulta que regrese el mismo resultado.

Select \* FROM engregan

WHERE clave <> 1000

1010 "BBBB800101" 5001 "2000-05-03 00:00:00.000" 528.00

1010 "BBBB800101" 5018 "2000-11-10 00:00:00.000" 667.00

1010 "BBBB800101" 5018 "2002-03-29 00:00:00.000" 523.00

Total: 129

**SQL  
select \* from entregan,materiales**

1000 "AAAA800101" 5000 "1998-07-08 00:00:00.000" 165.00 1000 "Varilla 3/16" 100.00

1000 "AAAA800101" 5019 "1999-08-08 00:00:00.000" 254.00 1000 "Varilla 3/16" 100.00

1000 "AAAA800101" 5019 "2000-04-06 00:00:00.000" 7.00 1000 "Varilla 3/16" 100.00

Total 5,808

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

En terminos de ambas, porque no se específica como una una condición por lo cual es el total de combinaciones posible.

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

Select descripcion

From materiales, entregan

Where materiales.clave = entregan.clave AND entregan.fecha BETWEEN '2000/01/01' AND '2000/12/31'

"Varilla 3/16"

"Varilla 4/32"

"Varilla 4/32"

Total 28

¿Por qué aparecen varias veces algunas descripciones de material?

Porque el material puede ser entregado varias veces.

**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?**

"Arena"

"Block"

"Cantera rosa"

Total: 22

Ya no se repiten los que fueron entregados varias veces.

**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 Entregan.Numero, Proyectos.Denominacion, Entregan.Fecha, Entregan.Cantidad

FROM Entregan, Proyectos

WHERE Entregan.Numero = Proyectos.Numero

ORDER BY Entregan.Numero, Entregan.Fecha

5000 "Vamos Mexico" "1998-07-08 00:00:00.000" 165.00

5000 "Vamos Mexico" "2000-03-05 00:00:00.000" 177.00

5000 "Vamos Mexico" "2002-03-12 00:00:00.000" 382.00

5001 "Aztecón" "1999-11-05 00:00:00.000" 43.00

Total:132

**SELECT \* FROM productos where Descripcion LIKE 'Si%'  
  
¿Qué resultado obtienes?**

1120 "Sillar rosa" 100.00

1130 "Sillar gris" 110.00

Total: 2

Todos los que empiezan con Si

**Explica que hace el símbolo '%'.**

Para indicar que puede ir antes, despues, o en ambos lados cualquier otro conjunto de caracteres, pero que debe ir a fuerzas lo indicado **¿Qué sucede si la consulta fuera : LIKE 'Si' ?**

Tendría que encontrar algo que fuera exactamente a eso. **¿Qué resultado obtienes?**

0, ya que no hay. **Explica a qué se debe este comportamiento.**

Como hago mención anteriormente, este tiene que estar obligado a ser solo la palabra Si.

DECLARE @foo varchar(40);  
DECLARE @bar varchar(40);  
SET @foo = '¿Que 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 variables. **¿Cuál es la función de @foo?**

Es una variable del tipo varchar, que tiene un string. **¿Que realiza el operador SET?**

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]%';

Muestra los RFC de la tabla entregan que empiezen entre A y D y los carecteres sucesores no son importantes.

"AAAA800101"

"AAAA800101"

"AAAA800101"

Total: 72

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

"BBBB800101"

"BBBB800101"

"BBBB800101"

Total 114

Regresa el complemento de todos los RFC que empiezan con A y tienen después cualquier conjunto de caracteres.

SELECT Numero FROM Entregan WHERE Numero LIKE '\_\_\_6';

5016

5016

5006

5006

Regresa todos los números de proyectos que tienen 3 cuales sean dígitos primero y el 4 es un 6.

**¿Cómo filtrarías rangos de fechas?**

Se puede dar el formato con SET, aunque en la consulta de fechas yo lo realicé de la manera siguiente:

BETWEEN '2000/01/01' AND '2000/12/31'

**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] )**

"AAAA800101" 165.00 "1998-07-08 00:00:00.000" 5000

"CCCC800101" 582.00 "2001-07-29 00:00:00.000" 5002

"AAAA800101" 86.00 "1999-01-12 00:00:00.000" 5008

Total 16  
  
¿Qué hace la consulta?

Regresa los campos que el numero sea entre 5000 y 5010 y que existan en la tabla de proveedores, tales que su razón social empiecen con La.  
¿Qué función tiene el paréntesis ( ) después de EXISTS?

Crear una SUB Consulta

**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 Entregan.RFC IN (SELECT RFC From Proveedores WHERE RazonSocial LIKE 'La%') AND Numero Between 5000 and 5010

**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.**

SELECT Materiales.Descripcion

FROM Materiales

WHERE Materiales.Clave = ANY (SELECT Clave FROM Entregan WHERE Cantidad < 100 )

**¿Qué hace la siguiente sentencia? Explica por qué.**  
  
**SELECT TOP 2 \* FROM Proyectos**

5000 "Vamos Mexico"

5001 "Aztecón"

Regresa los dos primeros registros, ya que el TOP se refiere a primeros el numero es hasta que número de registro.

**¿Qué sucede con la siguiente consulta? Explica por qué.**  
  
**SELECT TOP Numero FROM Proyectos**

Genera error, ya que como parametro espera un número.

**¿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 Entregan.Cantidad, Entregan.Clave, (Entregan.Cantidad\*(Materiales.Costo))+(Entregan.Cantidad\*Materiales.Costo\*(Materiales.PorcentajeImpuesto/100)) as 'Total'

From Entregan, Materiales

Where Entregan.Clave = Materiales.Clave

165.00 1000 16830.0000000000

254.00 1000 25908.0000000000

7.00 1000 714.0000000000

Total:132

**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.**

1.

CREATE VIEW PagosTotal AS

Select Entregan.Cantidad, Entregan.Clave, (Entregan.Cantidad\*(Materiales.Costo))+(Entregan.Cantidad\*Materiales.Costo\*(Materiales.PorcentajeImpuesto/100)) as 'Total'

From Entregan, Materiales

Where Entregan.Clave = Materiales.Clave

2.

CREATE VIEW RFCentreAD AS

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

3.

CREATE VIEW Descripcionentr2000 AS

Select Materiales.Descripcion

From materiales, entregan

Where materiales.clave = entregan.clave AND entregan.fecha BETWEEN '2000/01/01' AND '2000/12/31'

4.

CREATE VIEW CLAVES14501300 AS

SELECT \*

FROM Entregan

Where clave = 1450 OR clave = 1300

5.

CREATE VIEW MATERIALESCLAVE1000

select \* from materiales  
where clave=1000

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

SELECT materiales.Clave, Materiales.Descripcion

FROM Materiales, entregan, Proyectos

WHERE Materiales.Clave = Entregan.Clave AND Proyectos.Numero = Entregan.Numero and Proyectos.Denominacion='Mexico sin ti no estamos completos'

1030 "Varilla 4/33"

1230 "Cemento"

1430 "Pintura B1022"

Total: 3.

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

SELECT Materiales.Clave, Materiales.Descripcion

FROM Materiales, Entregan, Proveedores

WHERE Materiales.Clave = Entregan.Clave AND Entregan.RFC = Proveedores.RFC AND Proveedores.RazonSocial= 'Acme tools'

No regresa, porque ese proveedor no existe.

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

SELECT Entregan.RFC, AVG(Cantidad) AS promedio

FROM Entregan

WHERE Entregan.Fecha BETWEEN '2000/01/01' AND '2000/12/31'

GROUP BY RFC

HAVING AVG(Cantidad)>=300;

"BBBB800101" 392.400000

"CCCC800101" 372.000000

"DDDD800101" 314.333333

Total: 7

**El Total entregado por cada material en el año 2000.**

SELECT Entregan.Clave, SUM(Entregan.Cantidad) AS Total

FROM Entregan

WHERE Entregan.Fecha BETWEEN '2000/01/01' AND '2000/12/31'

GROUP BY Entregan.Clave

1000 7.00

1010 1195.00

1030 295.00

Total 22:  
  
**La Clave del material más vendido durante el 2001. (se recomienda usar una vista intermedia para su solución)**

CREATE VIEW Entregas2001 as

SELECT Entregan.Clave, SUM(Entregan.Cantidad) AS Total

FROM Entregan

WHERE Entregan.Fecha BETWEEN '2001/01/01' AND '2001/12/31'

GROUP BY Entregan.Clave

SELECT TOP 1 \*

FROM Entregas2001

1020 1060.00

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

**SELECT \***

**FROM Materiales**

**Where Materiales.Descripcion LIKE '%ub%'**

1180 "Recubrimiento P1001" 200.00 2.36

1190 "Recubrimiento P1010" 220.00 2.38

1200 "Recubrimiento P1019" 240.00 2.40

Total: 12

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

SELECT Proyectos.Denominacion, SUM((Entregan.Cantidad\*Materiales.Costo)\*(1+(Materiales.PorcentajeImpuesto/100))) AS Total

FROM Entregan, Proyectos, Materiales

WHERE Entregan.Clave = Materiales.Clave AND Proyectos.Numero = Entregan.Numero

GROUP BY Proyectos.Denominacion

"Ampliación de la carretera a la huasteca" 742461.1940000000

"Aztecón" 150200.2190000000

"CIT Campeche" 161603.0820000000

Total: 20

**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 TELEVISA AS

SELECT Proyectos.Denominacion, Proveedores.RFC, Proveedores.RazonSocial

FROM Proyectos, Entregan, Proveedores

WHERE Proveedores.RFC = Entregan.RFC AND Entregan.Numero = Proyectos.Numero AND Proyectos.Denominacion = 'Televisa en acción'

CREATE VIEW COAHUILA AS

SELECT Proyectos.Denominacion, Proveedores.RFC, Proveedores.RazonSocial

FROM Proyectos, Entregan, Proveedores

WHERE Proveedores.RFC = Entregan.RFC AND Entregan.Numero = Proyectos.Numero AND Proyectos.Denominacion = 'Educando en Coahuila'

SELECT \*

FROM TELEVISA

WHERE TELEVISA.RFC NOT IN (SELECT RFC FROM COAHUILA)

"Televisa en acción" "CCCC800101" "La Ferre"

"Televisa en acción" "CCCC800101" "La Ferre"

"Televisa en acción" "DDDD800101" "Cecoferre"

Total 6

**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 Proyectos.Denominacion, Proveedores.RFC, Proveedores.RazonSocial

FROM Entregan, Proyectos, Proveedores

WHERE Proveedores.RFC = Entregan.RFC AND Entregan.Numero = Proyectos.Numero AND Proyectos.Denominacion='Televisa en acción' AND Entregan.RFC NOT IN ( SELECT RFC FROM Entregan, Proyectos WHERE Entregan.Numero = Proyectos.Numero AND Proyectos.Denominacion='Educando en Coahuila')

"Televisa en acción" "CCCC800101" "La Ferre"

"Televisa en acción" "CCCC800101" "La Ferre"

"Televisa en acción" "DDDD800101" "Cecoferre"

Total 6

      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.  
  
**Reto: Usa solo el operador NOT IN en la consulta anterior (No es parte de la entrega).**

SELECT Materiales.Descripcion, Materiales.Costo

FROM Materiales, Entregan

WHERE Entregan.Clave = Materiales.Clave AND Entregan.RFC IN (

SELECT RFC

FROM TELEVISA

where Televisa.RFC IN(

SELECT RFC

FROM COAHUILA

)

**"Varilla 3/16" 100.00**

**"Varilla 3/16" 100.00**

**"Varilla 3/16" 100.00**

**"Ladrillos rojos" 50.00**

**Total 33**

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

create view contador as

SELECT Entregan.Clave, COUNT(Entregan.Clave) AS 'Veces', SUM(Cantidad) as 'Totally'

FROM Entregan, Materiales

WHERE Entregan.Clave = Materiales.Clave

GROUP BY Entregan.Clave

SELECT Materiales.descripcion, contador.Veces, Totally\*Materiales.Costo As Costo

FROM contador, Materiales

Where Materiales.Clave = contador.Clave

**Muchas de estas consultas requieren la utilización de funciones agregadas...**