



## **Construcción de Software y Toma de Decisiones (TC2005B)**

### **Laboratorio 20: Consultas en SQL**

**Profesores:** Ricardo Cortés Espinosa Eduardo Daniel Juárez Pineda

Rafael Hinojosa López - A01705777

Jorge Alan Ramírez Elías A01710350

Felipe Gabriel Yépez Villacreses - A01658002

Andrea Piñeiro Cavazos - A01705681

Joaquín Zermelo Saldaña – A01610399

Mariana Soto Ochoa - A01702593

## Laboratorio 20

### 1. Consulta de un tabla completa

```
SELECT * FROM Materiales
```

	Clave	Descripcion	Costo
1	1000	Varilla 3/16	100.00
2	1010	Varilla 4/32	115.00
3	1020	Varilla 3/17	130.00

La tabla contiene 44 tuplas.

### 2. Selección

```
SELECT *
```

```
FROM Materiales
```

```
WHERE Clave = 1000
```

	Clave	Descripcion	Costo
1	1000	Varilla 3/16	100.00

Regresa 1 renglón.

### 3. Proyección

```
SELECT Clave, RFC, Fecha
```

```
FROM Entregan
```

	Clave	RFC	Fecha
1	1000	AAAA800101	1998-07-08 00:00:00.000
2	1010	BBBB800101	2000-05-03 00:00:00.000
3	1020	CCCC800101	2001-07-29 00:00:00.000
4	1030	DDDD800101	1999-02-21 00:00:00.000

132 renglones.

### 4. Reunión Natural

```
SELECT *
```

```
FROM Materiales, Entregan
```

WHERE Materiales.Clave = Entregan.Clave

	Clave	Descripcion	Costo	Clave	RFC	Numero	Fecha	Cantidad
1	1000	Varilla 3/16	100.00	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00
2	1000	Varilla 3/16	100.00	1000	AAAA800101	5019	1999-08-08 00:00:00.000	254.00
3	1000	Varilla 3/16	100.00	1000	AAAA800101	5019	2000-04-06 00:00:00.000	7.00

132 renglones.

Si un material no se ha entregado no se muestra en la consulta porque un constraint es que no se muestren materiales con 0 elementos en la tabla Entregan.

5.

```
SELECT *
FROM Entregan, Proyectos
WHERE Entregan.Numero <= Proyectos.Numero
```

	Clave	RFC	Numero	Fecha	Cantidad	Numero	Denominacion
1	1000	AAAA800101	5000	1998-07-08 00:00:00.000	165.00	5000	Vamos Mexico
2	1200	EEEE800101	5000	2000-03-05 00:00:00.000	177.00	5000	Vamos Mexico
3	1400	AAAA800101	5000	2002-03-12 00:00:00.000	382.00	5000	Vamos Mexico

1188 renglones.

6. --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

-- Misma consulta sin union con OR

```
SELECT *
FROM Entregan
WHERE Clave = 1450 OR Clave = 1300
```

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

3 renglones.

7. intersección

```
SELECT Clave
FROM Entregan
WHERE numero = 5001
INTERSECT
SELECT Clave
FROM Entregan
WHERE numero = 5018
```

	clave
1	1010

1 renglon

8. Diferencia

```
SELECT *
FROM Entregan
WHERE Clave != 1000
```

	Clave
1	1010
2	1020
3	1030

129 renglones

9. 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 renglones = 44 renglones de Materiales x 132 renglones de Entregan

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

Para cada renglón de Entregan se asignan todos los renglones de materiales por consecuente el número de renglones totales será igual al producto de todos los renglones de cada tabla.

## 10. Construcción de consultas a partir de una especificación

```
SET DATEFORMAT dmy
SELECT Descripcion, Fecha
FROM Entregan,
WHERE Fecha <= '31/12/2000' AND Fecha >= '01/01/2000'
```

	Descripcion	Fecha
1	Varilla 3/16	2000-05-03 00:00:00.000
2	Varilla 4/32	2000-05-03 00:00:00.000
3	Varilla 3/17	2000-05-03 00:00:00.000

1232 renglones

### ¿Por qué aparecen varias veces algunas descripciones de material?

Por que pueden ser entregados varias veces a lo largo del año

## 11. Distinct

```
SET DATEFORMAT dmy
SELECT DISTINCT Descripcion, Fecha
FROM Entregan,
WHERE Fecha <= '31/12/2000' AND Fecha >= '01/01/2000'
```

	Descripcion	Fecha
1	Arena	2000-01-02 00:00:00.000
2	Arena	2000-02-03 00:00:00.000
3	Arena	2000-02-04 00:00:00.000
4	Arena	2000-02-05 00:00:00.000

1134 renglones.

### ¿Qué resultado obtienes en esta ocasión?

Los mismos pero ahora el conjunto de descripción y fecha son únicos.

## 12. Ordenamientos

```
SELECT P.Numero, Denominacion, Fecha, Cantidad
FROM Proyectos as P, Entregan
ORDER BY P.Numero ASC, Fecha DESC
```

	Numero	Denominacion	Fecha	Cantidad
1	5000	Vamos Mexico	2003-09-01 00:00:00.000	270.00
2	5000	Vamos Mexico	2003-08-05 00:00:00.000	549.00
3	5000	Vamos Mexico	2003-06-28 00:00:00.000	368.00

2640 renglones

### 13. Uso de expresiones

### 14. Operadores de Cadena

```
SELECT *  
FROM Materiales  
WHERE Descripcion LIKE 'Si%'
```

	Clave	Descripcion	Costo
1	1120	Sillar rosa	100.00
2	1130	Sillar gris	110.00

2 renglones.

Explica que hace el símbolo '%'.

El símbolo % indica que en su lugar pueden o no ir más caracteres para completar a los otros caracteres que están junto a él.

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

Regresaría todas las descripciones iguales a "Si". No hay ninguna con esa descripción.

Explica a qué se debe este comportamiento.

Este comportamiento se debe a que LIKE buscará un patrón en el campo escogido. Si se agrega una cadena de caracteres sin '%', se buscará la cadena como está escrita, pero si no, entonces se buscará cualquier patrón con la cadena escrita con '%' simulando la existencia o no de más caracteres alfanuméricos.

¿Qué resultado obtienes de ejecutar el siguiente código?

Obtengo la cadena de caracteres: "¿Que resultadoobtienes?¿¿¿???"

¿Para qué sirve DECLARE?

Para declarar variables en sql server.

¿Cuál es la función de @foo?

Es una variable que guarda una cadena de caracteres.

¿Que realiza el operador SET?

Le asigna valor a una variable.

Explica el significado, función y resultado de las siguientes consultas:

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

Resultado:

	RFC
1	AAAA800101
2	BBBB800101
3	AAAA800101

Función y significado:

Buscar los RFC que empiecen con letras entre A y B y que pueden o no contener más caracteres después.

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

Resultado:

99	CCCC800101
100	DDDD800101
101	EEEE800101

Función y significado:

Buscar todos los RFC que no (^) empiezan con A y que pueden o no contener más texto después.

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

Resultado:

	Numero
1	5006
2	5016
3	5006

Función y significado:

Buscar todos los números que contienen 3 caracteres y un 6. Cada caracter es representado por un guión bajo.

## Operadores lógicos

```
SELECT Clave, RFC, Numero, Fecha, Cantidad
```

```
FROM Entregan
```

```
WHERE Numero Between 5000 AND 5010;
```

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

Para especificar un rango en las fechas:

```
set dateformat dmy;
```

```
SELECT Clave, RFC, Numero, Fecha, Cantidad
```

```
FROM Entregan
```

WHERE Fecha Between '01/01/2000' AND '31/12/2000';

	Clave	RFC	Numero	Fecha	Cantidad
1	1010	BBBB800101	5001	2000-05-03 00:00:00.000	528.00
2	1050	FFFF800101	5005	2000-10-14 00:00:00.000	503.00
3	1060	GGGG800101	5006	2000-05-04 00:00:00.000	324.00

```

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?

La consulta proyecta el RFC, la Cantidad, Fecha y Numero de la tabla Entrega donde el Numero se encuentra entre 5000 y 5010 y a su vez pertenezca a una tupla de la tabla proveedores donde la razón social empiece con "La".

	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

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

Indicar dónde debe existir la columna especificada en WHERE. En este caso se realizó una subconsulta para buscar dentro de la tabla Proveedores.

Realiza el query con 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
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
4	CCCC800101	699.00	2001-09-10 00:00:00.000	5010

Usando ALL, SOME o ANY



```

SELECT RFC,Cantidad, Fecha,Numero
FROM Entregan
WHERE Numero Between 5000 and 5010
AND RFC = ANY ( SELECT RFC
                FROM Proveedores
                WHERE RazonSocial LIKE 'La%' and Entregan.RFC = Proveedores.RFC )

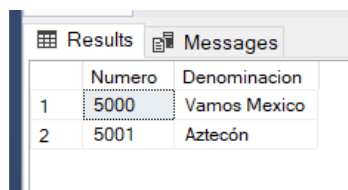
```

	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
4	CCCC800101	466.00	2000-08-06 00:00:00.000	5009
5	CCCC800101	523.00	2002-05-07 00:00:00.000	5009
6	CCCC800101	699.00	2001-09-10 00:00:00.000	5010

Operador TOP

¿Qué hace la siguiente sentencia? Explica por qué.

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



	Numero	Denominacion
1	5000	Vamos Mexico
2	5001	Aztecón

Regresa los 2 proyectos de hasta arriba

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

```
SELECT TOP Numero FROM Proyectos
```

El código no corre ya que tiene un sintaxis invalido, esto debido a que se tiene que especificar un rango.

### Modificando la estructura de una tabla existente

Agrega a la tabla materiales la columna PorcentajImpuesto con la instrucción:

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

A fin de que los materiales tengan un impuesto, les asignaremos impuestos ficticios basados en sus claves con la instrucción:

```
UPDATE materiales SET PorcentajImpuesto = 2*clave/1000;
```

esto es, a cada material se le asignará un impuesto igual al doble de su clave dividida entre diez.

Revisa la tabla de materiales para que compruebes lo que hicimos anteriormente.

```
SELECT * FROM materiales
```

	Clave	Descripcion	Costo	PorcentajImpuesto
1	1000	Varilla 3/16	100.00	2.00
2	1010	Varilla 4/32	115.00	2.02
3	1020	Varilla 3/17	130.00	2.04
4	1030	Varilla 4/33	145.00	2.06

¿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 M.Clave, (Cantidad * (Costo + Costo * PorcentajImpuesto/100)) as 'Total'
FROM Materiales M, Entregan E
WHERE M.Clave = E.Clave
```

	Clave	Total
1	1000	16830.0000000000
2	1000	25908.0000000000
3	1000	714.0000000000
4	1010	61946.5440000000
5	1010	78254.4410000000

Para mostrarlo en otra columna usaría ALTER TABLE para asignar una nueva columna de importe, después usaría UPDATE y SET para darle los valores a cada fila.

### Creación de vistas

Crea 5 vistas usando consultas previas:

```
CREATE view Entrega(RFC, Cantidad, Fecha, Numero)
AS 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] )
```

```
CREATE view Num6(Numero)
```

```
AS SELECT Numero FROM Entregan WHERE Numero LIKE '___6';
```

```
CREATE view RFCAB(RFCAB)
```

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

```
CREATE view DescSI(Clave, Descripcion, Costo, PorcentajeImp)
```

```
AS SELECT *
```

```
FROM Materiales
```

```
WHERE Descripcion LIKE 'Si%'
```

```
CREATE view NoA(RFC)
```

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

-----

### Ejercicios

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

```
SELECT M.Clave, M.Descripcion
```

```
FROM Materiales as M, Entregan as E
```

```
WHERE M.Clave = E.Clave
```

```
AND E.Numero = (Select P.Numero
```

```
FROM Proyectos as P
```

```
WHERE Denominacion = 'Mexico sin ti no estamos completos')
```

	Clave	Descripcion
1	1030	Varilla 4/33
2	1230	Cemento
3	1430	Pintura B1022

3 renglones.

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

```
SELECT M.Clave, M.Descripcion
```

```
FROM Materiales as M, Entregan as E
```

```
WHERE M.Clave = E.Clave
```

```
AND E.RFC = (SELECT PR.RFC
```

```
FROM Proveedores as PR
WHERE RazonSocial = 'Acme tools');
```

Clave	Descripcion
-------	-------------

0 renglones porque no existe dicho proveedor.

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

```
SET DATEFORMAT dmy;
SELECT RFC
FROM Entregan
WHERE Fecha BETWEEN '01/01/2000' AND '31/12/2000'
GROUP BY RFC
HAVING AVG(Cantidad) >= 300
```

	RFC
1	BBBB800101
2	CCCC800101
3	DDDD800101

7 renglones.

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

```
SET DATEFORMAT dmy;
SELECT Clave, SUM(Cantidad)
FROM Entregan
WHERE Fecha BETWEEN '01/01/2000' AND '31/12/2000'
GROUP BY Clave
```

	Clave	Suma
1	1000	7.00
2	1010	1195.00
3	1030	295.00

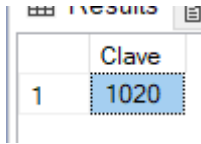
22 renglones.

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

```
Create view Sumas (Clave, Cantidad_Vendida)
AS (SELECT Clave, SUM(Cantidad)
FROM Entregan
WHERE Fecha BETWEEN '01/01/2001' AND '31/12/2001')
```

GROUP BY clave)

```
SELECT Clave
FROM Sumas
WHERE Cantidad_Vendida = (SELECT MAX(Cantidad_Vendida)
                           FROM Sumas)
```

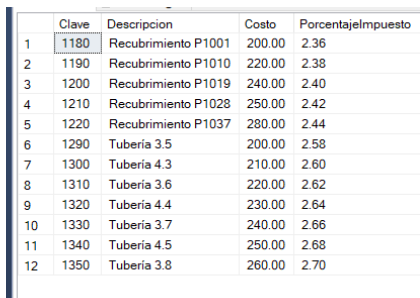


	Clave
1	1020

1 renglón

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

```
SELECT *
FROM Materiales
WHERE Descripcion LIKE '%ub%'
```

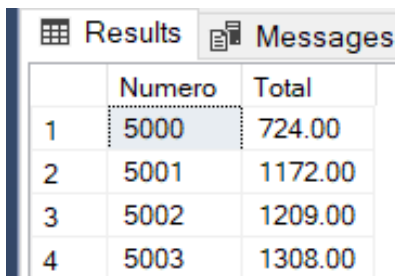


	Clave	Descripcion	Costo	PorcentajeImpuesto
1	1180	Recubrimiento P1001	200.00	2.36
2	1190	Recubrimiento P1010	220.00	2.38
3	1200	Recubrimiento P1019	240.00	2.40
4	1210	Recubrimiento P1028	250.00	2.42
5	1220	Recubrimiento P1037	280.00	2.44
6	1290	Tubería 3.5	200.00	2.58
7	1300	Tubería 4.3	210.00	2.60
8	1310	Tubería 3.6	220.00	2.62
9	1320	Tubería 4.4	230.00	2.64
10	1330	Tubería 3.7	240.00	2.66
11	1340	Tubería 4.5	250.00	2.68
12	1350	Tubería 3.8	260.00	2.70

12 renglones

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

```
SELECT Numero, SUM(Cantidad) as Total
FROM Entregan
Group by Número
```



	Numero	Total
1	5000	724.00
2	5001	1172.00
3	5002	1209.00
4	5003	1308.00

20 renglones

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

```
CREATE VIEW TeA(RFC, RazonSocial, Denominacion)
AS SELECT DISTINCT P.RFC, RazonSocial, Denominacion
FROM Entregan as E, Proveedores as P, Proyectos as pp
WHERE RazonSocial = 'Televisa en acción'
```

```
CREATE VIEW EeC(RFC, RazonSocial, Denominacion)
AS SELECT DISTINCT P.RFC, RazonSocial, Denominacion
FROM Entregan as E, Proveedores as P, Proyectos as pp
WHERE RazonSocial = 'Educando en Coahuila'
```

```
SELECT DISTINCT RFC, Denominacion, RazonSocial
FROM TeA
WHERE RFC NOT IN (SELECT RFC FROM EeC)
```

-No se encontraron resultados

*Denominación, RFC y RazonSocial de los proveedores que 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.RFC, RazonSocial, Denominacion
FROM Entregan as E, Proveedores as P, Proyectos as pp
WHERE RazonSocial = 'Televisa en acción'
AND P.RFC NOT IN (SELECT RFC
FROM Entregan
WHERE RazonSocial = 'Educando en Coahuila')
```

-No se encontraron resultados

*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 Costo, Descripcion
FROM Entregan as E, Materiales as M, Proyectos as P
WHERE M.Clave = E.Clave
AND E.Numero = P.Numero
AND Denominacion = 'Televisa en acción'
AND E.RFC IN (SELECT E.RFC
FROM Entregan as E, Proyectos as P
WHERE E.Numero = P.Numero)
```

AND Denominacion = 'Educando en Coahuila')

Results		Messages	
	Costo	Descripcion	
1	50.00	Ladrillos rojos	
2	34.00	Tepetate	

2 renglones.

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

```
SELECT Descripcion, COUNT(*) AS 'Número de veces entregado', SUM(Cantidad * (Costo
+ Costo * Porcentajelmpuesto/100)) AS 'Costo Total'
FROM Entregan E, Materiales M
WHERE E.Clave = M.Clave
GROUP BY E.Clave, Descripcion
```

	Descripcion	Número de veces entregado	Costo Total
1	Vanilla 3/16	3	43452.0000000000
2	Vanilla 4/32	3	201560.9140000000
3	Vanilla 3/17	3	141672.3360000000
4	Vanilla 4/33	3	94119.7320000000
5	Vanilla 3/18	3	220329.4720000000
6	Vanilla 4/34	3	217268.8000000000
7	Vanilla 3/19	3	206251.7640000000
8	Vanilla 4/35	3	213784.1270000000
9	Ladrillos rojos	3	62011.1200000000
10	Ladrillos grises	3	39553.8780000000