

Instituto Tecnológico de Estudios Superiores Monterrey

CAMPUS QUERÉTARO

Construcción de software y toma de decisiones

Ricardo Cortés Espinosa

Eduardo Daniel Juárez Pineda

Grupo 401

**Ejercicio: Consultas SQL**

**PRESENTA**

Leonardo Santiago Ramos Pérez - A 01707812

Fecha:

8 de septiembre de 2022

**Consulta de una tabla completa**

**Algebra relacional:**

Materiales

**SQL:**

Select \* From materiales

****

**Selección**

**Algebra relacional:**

SL {clave=1000} (materiales)

**SQL:**

select \* from materiales  
where clave=1000

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Proyección**

**Algebra relacional:**

PR {clave, rfc, fecha} (entregan)

**SQL:**

select clave, rfc, fecha from entregan

**Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media**

**Reunión Natural**

**Algebra relacional:**

entregan JN materiales

**SQL:**

**select \* from materiales,entregan**

where materiales.clave = entregan.clave

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

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

No pues solo mostrara los materiales entregados

**Reunión con criterio específico**

**Algebra relacional.**

entregan JN { entregan.numero <= proyectos.numero} proyectos

**SQL**

select \* from entregan,proyectos

where entregan.numero <= proyectos.numero

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Unión (se ilustra junto con selección)**

**Algebra relacional.**

SL{clave=1450}(entregan) UN SL{clave=1300}(entregan)

**SQL**

(select \* from entregan where clave=1450)

union

(select \* from entregan where clave=1300)

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

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

select \* from entregan where clave=1450 or clave = 1300

Interfaz de usuario gráfica

Descripción generada automáticamente

**Intersección (se ilustra junto con selección y proyección)**

**Algebra relacional**:

PR{clave}(SL{numero=5001}(entregan)) IN PR{clave}(SL{numero=5018}(entregan))

**SQL:**

select clave from entregan where numero=5001 AND clave IN (select clave from entregan where numero=5018)

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Diferencia (se ilustra con selección )**

**Algebra relacional.**

entregan - SL{clave=1000}(entregan)

**SQL**

SELECT \* FROM entregan

WHERE clave NOT IN

(SELECT clave FROM entregan WHERE clave = 1000)

Tabla

Descripción generada automáticamente con confianza baja

**Alterna**

(select \* from entregan)

EXCEPT

(select \* from entregan where clave=1000)

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Producto cartesiano**   
  
**Algebra relacional.**  
entregan X materiales  
  
**SQL**  
select \* from entregan,materiales

Interfaz de usuario gráfica, Texto, Aplicación

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

Se multiplican ambas tuplas (entregan - 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.

**SQL**

SELECT descripcion FROM materiales A, entregan E WHERE A.clave = E.clave AND fecha BETWEEN '2000/01/01' AND '2000/12/31'

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente  
  
¿Por qué aparecen varias veces algunas descripciones de material?

La consulta nos muestra las entregas realizadas en el 2000, mostrándonos las entregas de un mismo material más de una vez en ese año.

**Uso del calificador distinct**

Agrega la palabra distinct inmediatamente después de la palabra select a la consulta que planteaste antes.

**SQL**.

SELECT DISTINCT descripcion FROM materiales A, entregan E

WHERE A.clave = E.clave AND fecha BETWEEN '2000/01/01' AND '2000/12/31'

¿Qué resultado obtienes en esta ocasión?

Solo la descripción de los material entregados en el 2000

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**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.

**SQL.**

SELECT P.numero, denominacion,fecha,cantidad

FROM proyectos as P, entregan as E

Where P.numero = E.numero

GROUP BY p.numero

ORDER by fecha DESC;

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Operadores de Cadena**

**SQL**

SELECT \* FROM materiales where Descripcion LIKE 'Si%'

**¿Qué resultado obtienes?**

La descripción de solo los materiales que empiezan con la palabra Si.

**Explica que hace el símbolo '%'.**

Hace que tome los caracteres que siguen como válidos.

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

Solo aplicaría estrictamente la búsqueda en la palabra “SI”

**¿Qué resultado obtienes?**

No hay ninguna descripción con solo la palabra SI

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Operadores Lógico**

**SQL:**

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 )

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

**¿Qué hace la consulta?**

Selecciona el RFC, Cantidad, Fecha, y numero de entrega que se encuentre entre

5000 y 5010, además su RFC debe ser de proveedores donde su razón

social sea como ‘La seguido demás caracteres’

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

Prioriza la resolución de consulta, por lo que este entre paréntesis se realiza primero.

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

**SQL:**

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 );

Imagen que contiene Tabla

Descripción generada automáticamente

**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.**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

¿Qué hace la siguiente sentencia? Explica por qué.

**SELECT TOP 2 \* FROM Proyectos**

Toma los 2 primero registro de la tabla proyectos

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

**SELECT TOP Numero FROM Proyectos**

Toma el primer registro en la tabla proyectos

**Modificando la estructura de un tabla existente.**

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

**SQL:**

ALTER TABLE materiales ADD PorcentajeImpuesto NUMERIC(6,2);

UPDATE materiales SET PorcentajeImpuesto = 2\*clave/1000;

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**¿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?**

**SQL:**

SELECT M.descripcion,(cantidad \*( precio + impuesto)) as 'Importe'

FROM materiales M, entregan E

WHERE M.clave = E.clave;

Captura de pantalla de computadora

Descripción generada automáticamente con confianza media

**Creación de vistas**

**SQL:**  
CREATE VIEW ProyectoMexicoSinTi

AS SELECT M.clave, M.descripcion

FROM materiales M,entregan E, proyectos P

WHERE E.clave = M.clave AND P.numero = E.numero AND P.denominacion =

'México sin ti no estamos completos';

SELECT \* FROM proyectomexicosinti;

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

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

**SQL:**

CREATE VIEW matTubasa

AS SELECT M.clave, M.descripcion

FROM materiales M, entregan E, proveedores P

WHERE E.clave = M.clave AND P.RFC = E.RFC AND P.razonsocial = 'Tubasa';

SELECT \* FROM matTubasa

Interfaz de usuario gráfica, Aplicación, Icono

Descripción generada automáticamente

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

**SQL:**

SELECT M.descripcion, E.Cantidad

FROM materiales M JOIN entregan E USING (clave)

WHERE E.fecha BETWEEN '2000/01/01' AND '2000/12/31'

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

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

Imagen que contiene Texto

Descripción generada automáticamente

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

**SQL:**

SELECT \*

FROM materiales

WHERE descripcion LIKE '%ub%';

**Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente**

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

**SQL:**

SELECT denominacion, SUM(E.cantidad \* (M.precio + M.impuesto)) as 'Total a pagar'

FROM proyectos P, entregan E, materiales M

WHERE P.numero = E.numero AND M.clave = E.clave

GROUP BY denominacion

**Tabla

Descripción generada automáticamente**

**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).**

**SQL:**

CREATE VIEW Televisa

AS SELECT denominacion, PR.RFC, razonsocial

FROM proyectos P, proveedores PR, entregan E

WHERE E.numero = P.numero AND E.RFC = PR.RFC AND P.denominacion = ‘Televisa en

acción’;

**Imagen que contiene Diagrama

Descripción generada automáticamente**

CREATE VIEW EducandoCoahuila

AS SELECT denominacion, PR.RFC, razonsocial

FROM proyectos P, proveedores PR, entregan E

WHERE E.numero = P.numero AND E.RFC = PR.RFC AND P.denominacion = ‘Educando en

Coahuila’;

**Texto

Descripción generada automáticamente con confianza baja**

SELECT T.denominacion, T.RFC, T.razonsocial

FROM Televisa as T, EducandoCoahuila as C

WHERE T.RFC NOT IN (C.RFC)

GROUP BY T.razonsocial

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**

**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).**

**SQL:**

SELECT denominacion, PR.RFC, razonsocial

FROM proyectos P, proveedores PR, entregan E

WHERE E.numero = P.numero AND E.RFC = PR.RFC AND P.denominacion = ‘Televisa en

acción’ AND PR.RFC NOT IN (SELECT PR.RFC

FROM proyectos P, proveedores PR, entregan E

WHERE E.numero = P.numero

AND E.RFC = PR.RFC

AND P.denominacion = ‘Educando en Coahuila’ )

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

**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**

**SQL:**

SELECT M.descripcion, (precio + impuesto) as 'Costo material'

FROM proyectos P, proveedores PR, entregan E, materiales M

WHERE E.numero = P.numero AND E.RFC = PR.RFC AND E.clave = M.clave

AND P.denominacion = 'Televisa en acción' AND PR.RFC IN (SELECT PR.RFC FROM proyectos P, proveedores PR, entregan E WHERE E.numero = P.numero AND E.RFC = PR.RFC AND P.denominacion = 'Educando en Coahuila' )

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente