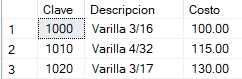
* ****
* **Construcción de Software y Toma de Decisiones (TC2005B)**
* **Laboratorio 20: Consultas en SQL**
* **Profesores:** [Ricardo Cortés Espinosa](mailto:rcortese@tec.mx) **,** [Eduardo Daniel Juárez Pineda](mailto:edjuarezp@tec.mx)
* [Rafael Hinojosa López](mailto:a01705777@itesm.mx) - A01705777
* [Jorge Alan Ramírez Elías](mailto:a01701350@itesm.mx) A01710350
* [Felipe Gabriel Yépez Villacreses](mailto:a01658002@itesm.mx) - A01658002
* [Andrea Piñeiro Cavazos](mailto:a01705681@itesm.mx) - A01705681
* [Joaquín Zermeño Saldaña](mailto:a01610399@itesm.mx) – A01610399
* [Mariana Soto Ochoa](mailto:a01702593@itesm.mx) - A01702593
* Campus Querétaro,
* Viernes 12 de marzo de 2021

**Laboratorio 20**

1. Consulta de un tabla completa

SELECT \* FROM Materiales



La tabla contiene 44 tuplas.

2. Selección

SELECT \*

FROM Materiales

WHERE Clave = 1000

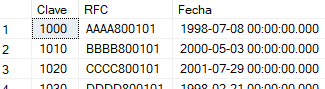


Regresa 1 renglón.

3. Proyección

SELECT Clave, RFC, Fecha

FROM Entregan



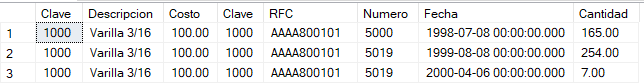
132 renlgones.

4. Reunión Natural

SELECT \*

FROM Materiales, Entregan

WHERE Materiales.Clave = Entregan.Clave



132 renglones.

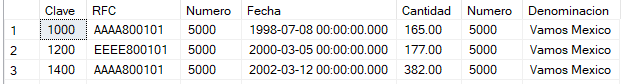
Si un material no se ha entregado no se muestra en la consulta porque un constraint es que no se muestren materiales con 0 elementos en la tabla Entregan.

5.

SELECT \*

FROM Entregan, Proyectos

WHERE Entregan.Numero <= Proyectos.Numero



1188 renglones.

6. --Unión (se ilustra junto con selección)

SELECT \*

FROM Entregan

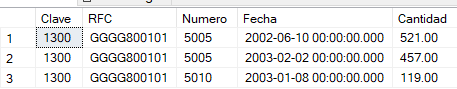
WHERE Clave = 1450

UNION

SELECT \*

FROM Entregan

WHERE Clave = 1300



-- Misma consulta sin union con OR

SELECT \*

FROM Entregan

WHERE Clave = 1450 OR Clave = 1300



3 renglones.

7. intersección

SELECT Clave

FROM Entregan

WHERE numero = 5001

INTERSECT

SELECT Clave

FROM Entregan

WHERE numero = 5018



1 renglon

8. Diferencia

SELECT \*

FROM Entregan

WHERE Clave != 1000

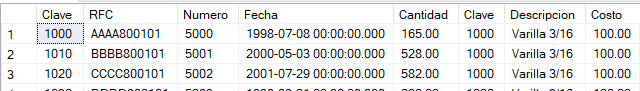


129 renglones

9. Producto Cartesiano

SELECT \*

FROM Entregan, Materiales



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'



1232 renglones

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

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

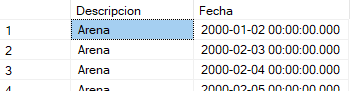
11. Distinct

SET DATEFORMAT dmy

SELECT DISTINCT Descripcion, Fecha

FROM Entregan,

WHERE Fecha <= '31/12/2000' AND Fecha >= '01/01/2000'



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



2640 renglones

13. Uso de expresiones

14. Operadores de Cadena

SELECT \*

FROM Materiales

WHERE Descripcion LIKE 'Si%'



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 si 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.

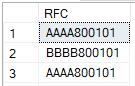
¿Que realiza el operador SET?

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:



Función y significado:

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

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

Resultado:

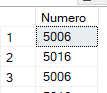


Función y significado:

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:



Función y significado:

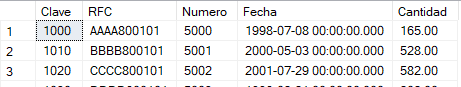
Buscar todos los números que contienen 3 caracteres y un 6. Cada caracter es representado por un guión bajo.

**Operadores lógicos**

SELECT Clave, RFC, Numero, Fecha, Cantidad

FROM Entregan

WHERE Numero Between 5000 AND 5010;



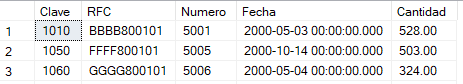
Para especificar un rango en las fechas:

set dateformat dmy;

SELECT Clave, RFC, Numero, Fecha, Cantidad

FROM Entregan

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



SELECT RFC, Cantidad, Fecha, Numero

FROM [Entregan]

WHERE [Numero] Between 5000 AND 5010 AND

EXISTS (SELECT [RFC]

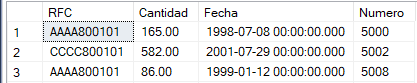
FROM [Proveedores]

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”.



¿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

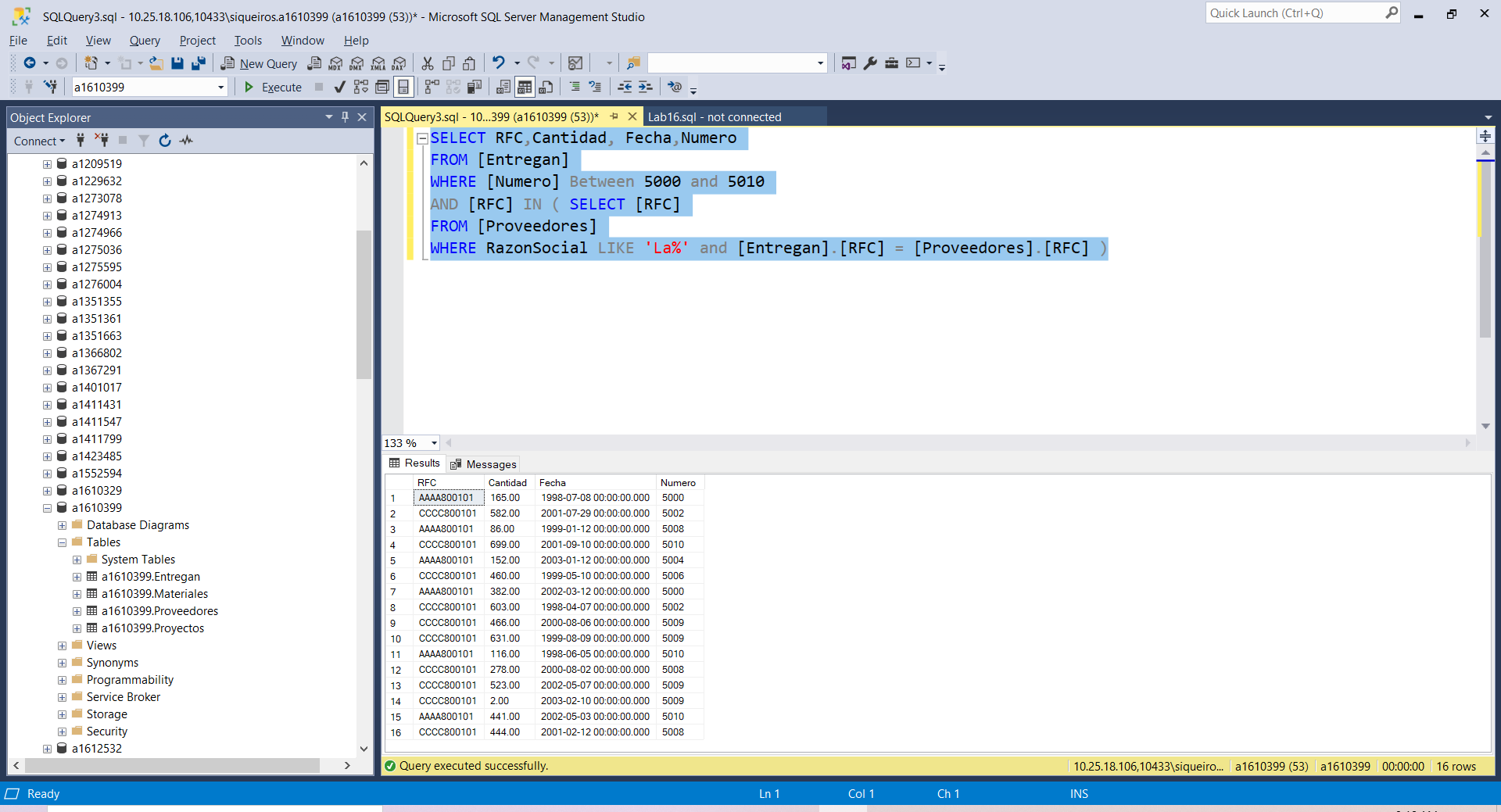
FROM [Entregan]

WHERE [Numero] Between 5000 and 5010

AND [RFC] IN ( SELECT [RFC]

FROM [Proveedores]

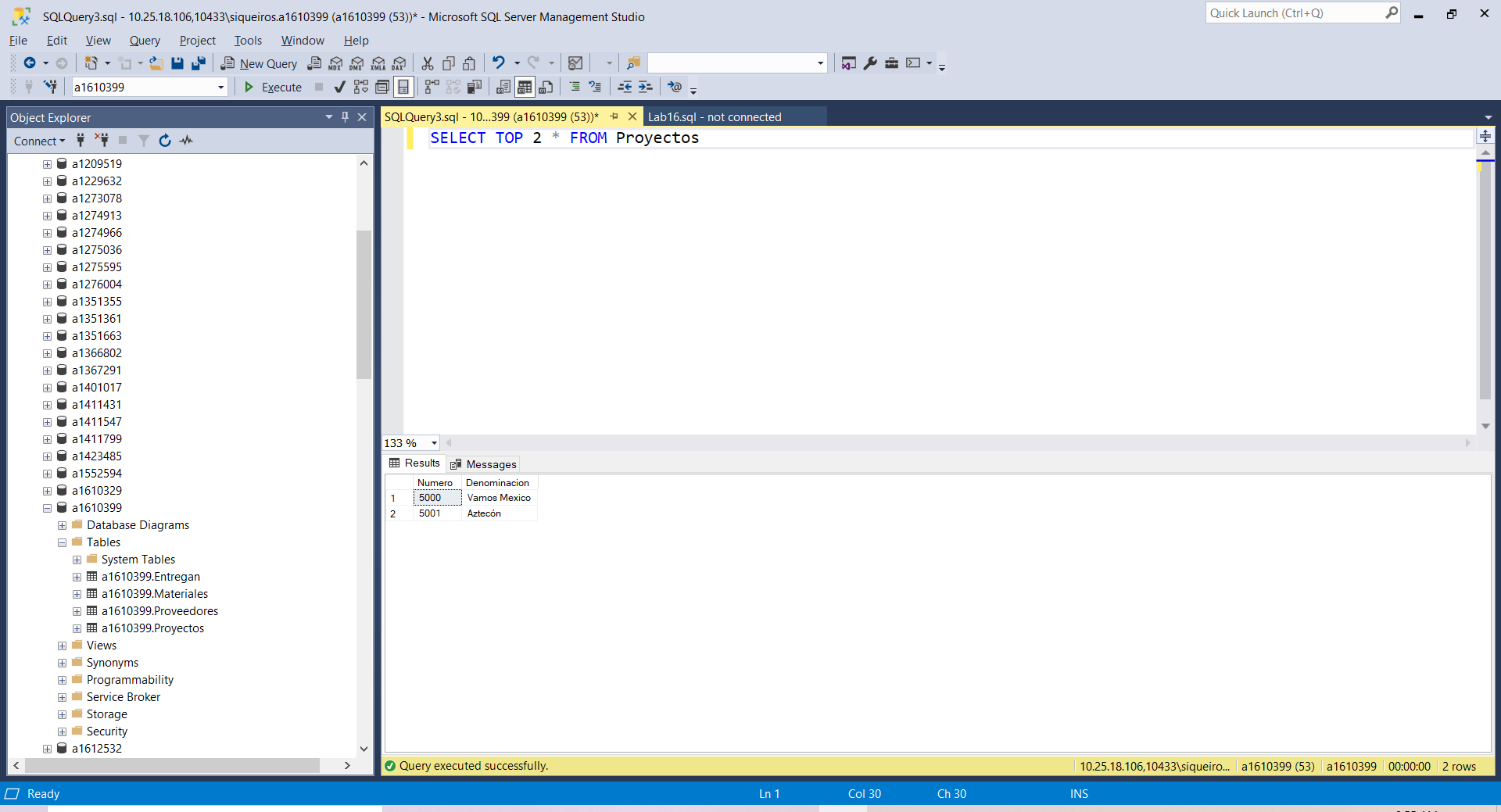
WHERE RazonSocial LIKE 'La%' and [Entregan].[RFC] = [Proveedores].[RFC] )



Operador TOP

¿Qué hace la siguiente sentencia? Explica por qué.

SELECT TOP 2 \* FROM Proyectos



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 PorcentajeImpuesto con la instrucción:

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

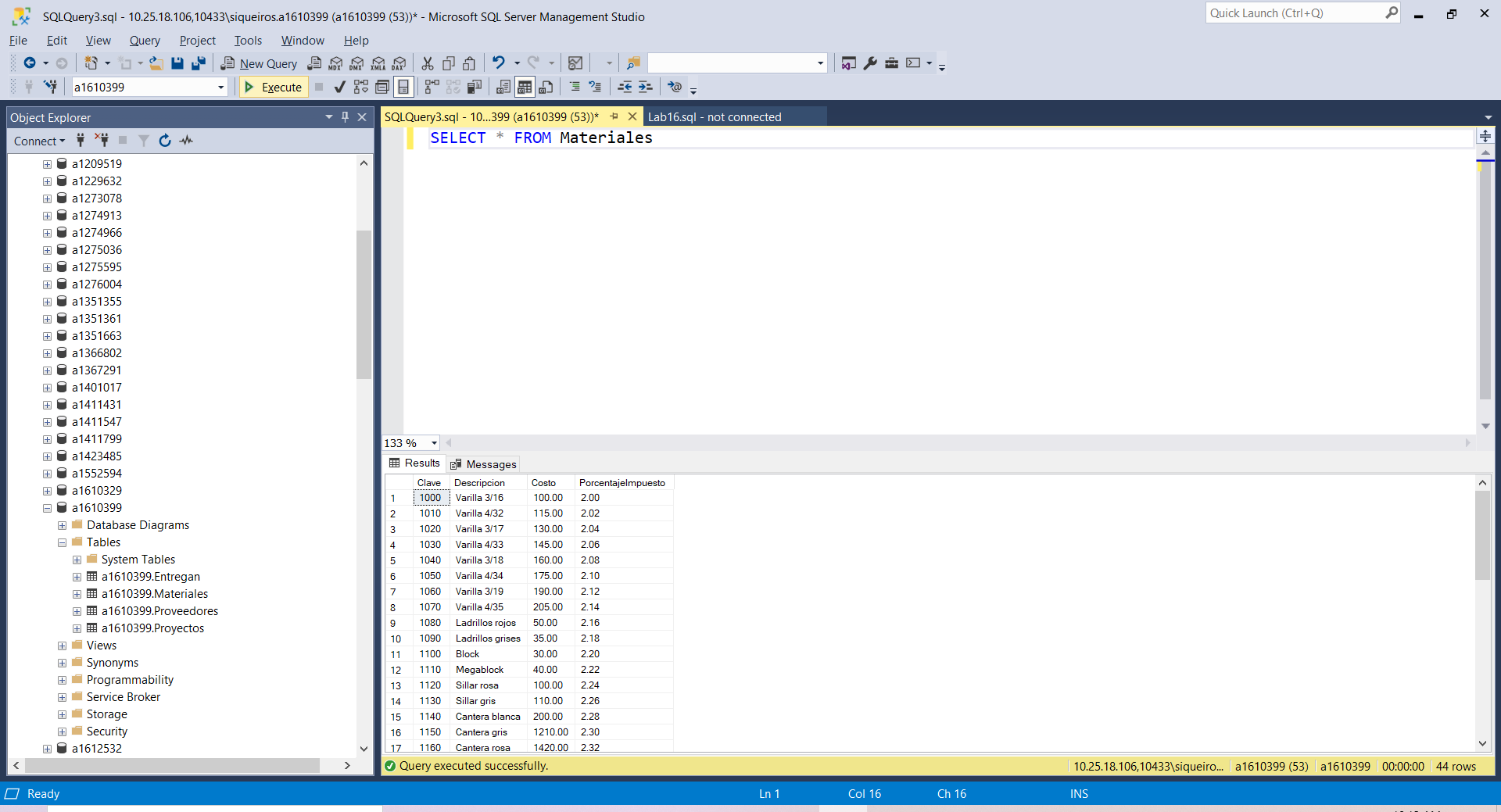
A fin de que los materiales tengan un impuesto, les asignaremos impuestos ficticios basados en sus claves con la instrucción:

UPDATE materiales SET PorcentajeImpuesto = 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



¿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?

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

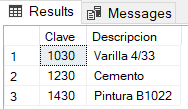
FROM Materiales as M, Entregan as E

WHERE M.Clave = E.Clave

AND E.Numero = (Select P.Numero

FROM Proyectos as P

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



3 renglones.

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

SELECT M.Clave, M.Descripcion

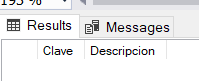
FROM Materiales as M, Entregan as E

WHERE M.Clave = E.Clave

AND E.RFC = (SELECT PR.RFC

FROM Proveedores as PR

WHERE RazonSocial = 'Acme tools');



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;

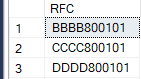
SELECT RFC

FROM Entregan

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

GROUP BY RFC

HAVING AVG(Cantidad) >= 300



7 renglones.

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

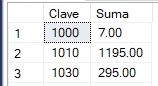
SET DATEFORMAT dmy;

SELECT Clave, SUM(Cantidad)

FROM Entregan

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

GROUP BY Clave



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)



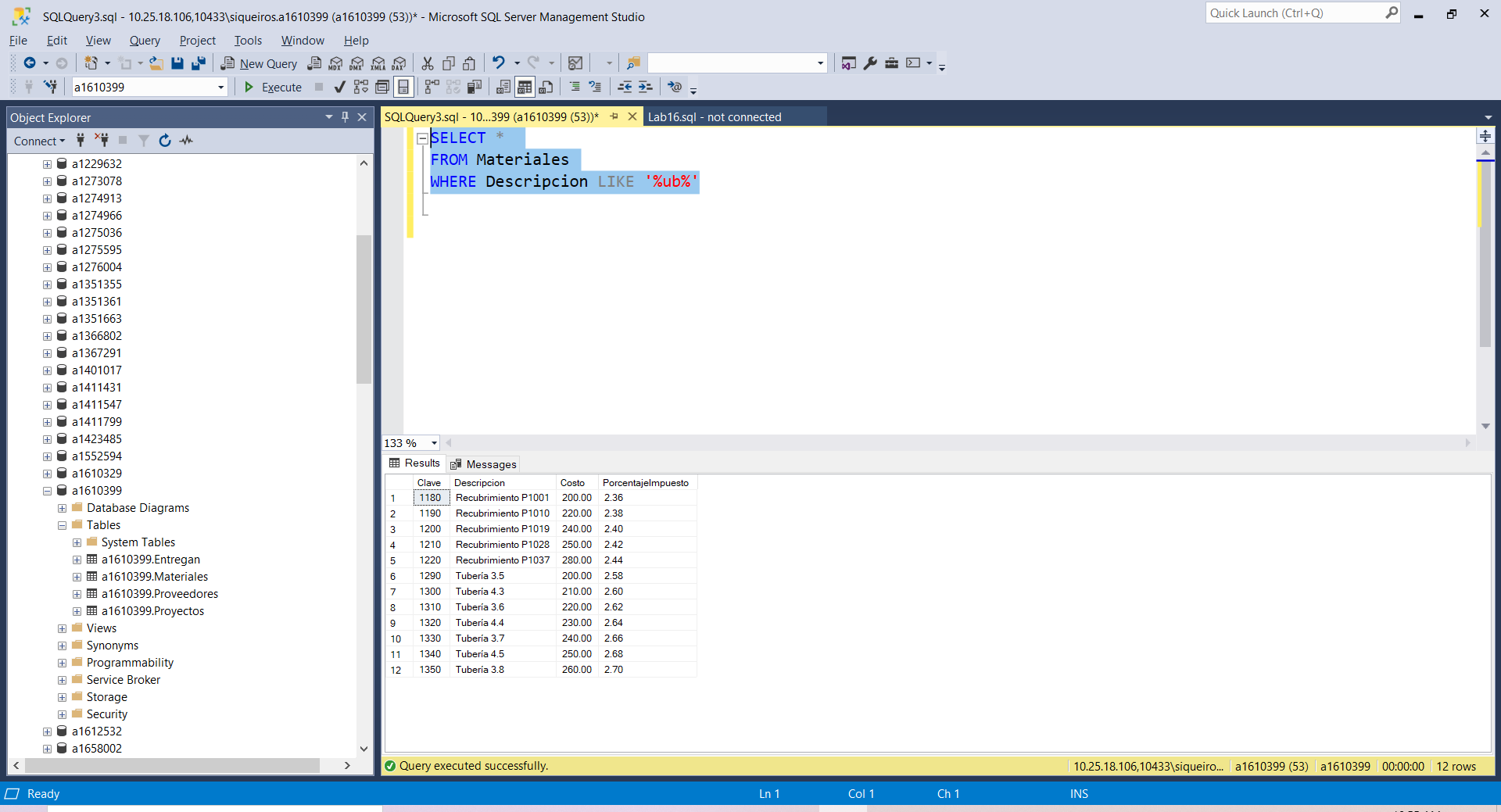
1 renglón

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

*SELECT \**

*FROM Materiales*

*WHERE Descripcion LIKE '%ub%'*

**

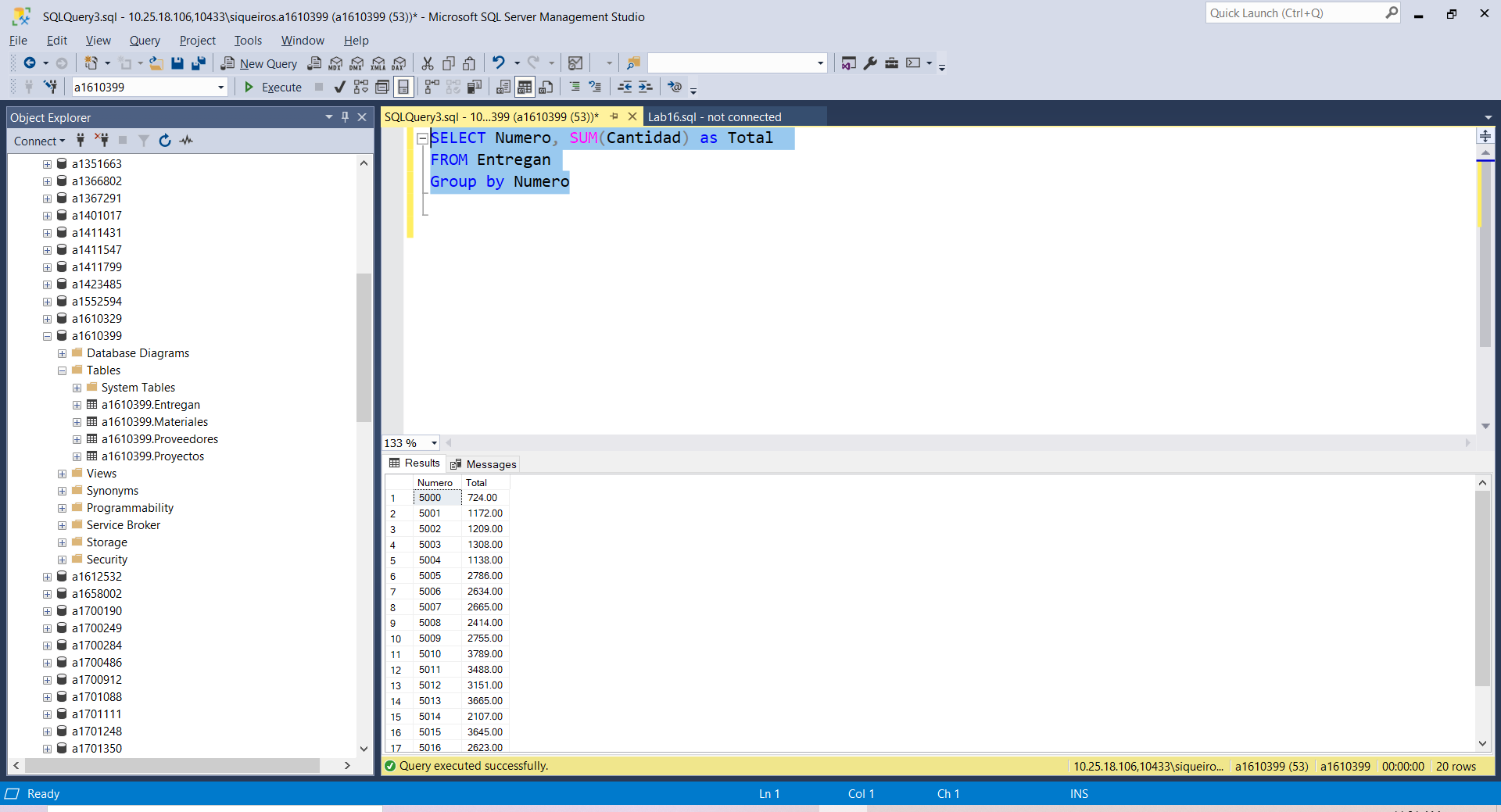
*12 renglones*

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

*SELECT Numero, SUM(Cantidad) as Total*

*FROM Entregan*

*Group by Número*

**

*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

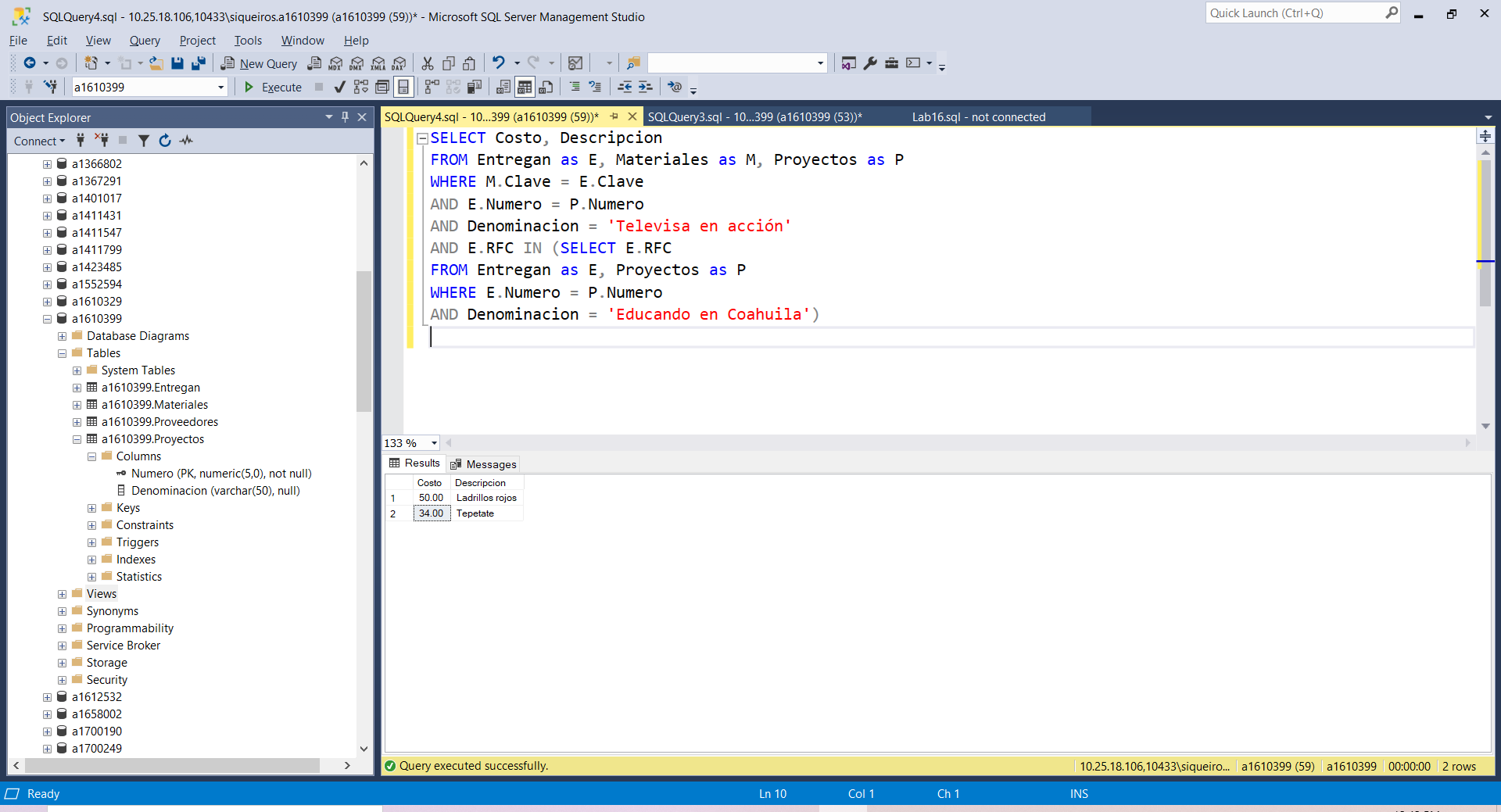
AND Denominacion = 'Televisa en acción'

AND E.RFC IN (SELECT E.RFC

FROM Entregan as E, Proyectos as P

WHERE E.Numero = P.Numero

AND Denominacion = 'Educando en Coahuila')



2 renglones.