



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

**Laboratorio 23: Manipulación de datos usando  
Stored Procedures**

**Profesores:** [Ricardo Cortés Espinosa](#) , [Eduardo Daniel Juárez Pineda](#)

Andrea Piñeiro Cavazos - A01705681

Campus Querétaro,

Lunes 22 de marzo de 2021

## Lab 23

Copia la siguiente sentencia en SQL Server Management Studio:

```
IF EXISTS (SELECT name FROM sysobjects
            WHERE name = 'creaMaterial' AND type = 'P')
    DROP PROCEDURE creaMaterial
GO

CREATE PROCEDURE creaMaterial
    @uclave NUMERIC(5,0),
    @udescpcion VARCHAR(50),
    @ucosto NUMERIC(8,2),
    @uimpuesto NUMERIC(6,2)
AS
    INSERT INTO Materiales VALUES(@uclave, @udescpcion, @ucosto, @uimpuest
o)
GO
```

En MySQL:

```
DROP PROCEDURE IF EXISTS creaMaterial;
DELIMITER //
CREATE PROCEDURE creaMaterial(
    IN clave INT(11),
    IN descripcion VARCHAR(40),
    IN precio FLOAT,
    IN impuesto FLOAT
)
BEGIN
    INSERT INTO materiales VALUES (clave, descripcion, precio, impuesto);
END //
```

### ¿Qué hace el primer bloque del código (bloque del IF)?

Revisa todos los objetos creados en la base de datos (sysobjects), y hace una consulta del método 'crearMateriales', con type = 'P' para asegurarse de que sea un procedimiento. Si dicho procedimiento existe, lo elimina.

### ¿Para que sirve la instrucción GO?

Go indica a SQL Server que es el final de una serie de instrucciones que deben ejecutarse como lote; el conjunto de instrucciones está formada desde el último comando GO o desde el comienzo del archivo.

### ¿Explica que recibe como parámetro este Procedimiento y qué tabla modifica?

Este procedimiento recibe clave, descripción, costo e impuesto del material que se quiere insertar en la tabla; y la tabla modificada es 'Materiales'. El procedimiento añade un registro más a la tabla, con los valores recibidos como parámetros.

Para ejecutar el Stored Procedure en MySQL:

```
CALL creaMaterial(5000,'Martillos Acme',250,15);
```

Verificar el efecto de la ejecución:

```
SELECT * FROM Materiales
```

Antes:

clave	descripcion	precio	impuesto
1000	Varilla 3/16	100	10
1010	Varilla 4/32	115	11.5
1020	Varilla 3/17	130	13
1030	Varilla 4/33	145	14.5
1040	Varilla 3/18	160	16
1050	Varilla 4/34	175	17.5
1060	Varilla 3/19	190	19
1070	Varilla 4/35	205	20.5
1080	Ladrillos rojos	50	5
1090	Ladrillos grises	35	3.5
1100	Block	30	3
1110	Megablock	40	4
1120	Sillar rosa	100	10
1130	Sillar gris	110	11
1140	Cantera blanca	200	20
1150	Cantera gris	1210	121
1160	Cantera rosa	1420	142
1170	Cantera amarilla	230	23
1180	Recubrimiento P1001	200	20
1190	Recubrimiento P1010	220	22
1200	Recubrimiento P1019	240	24
1210	Recubrimiento P1028	250	25
1220	Recubrimiento P1037	280	28
1230	Cemento	300	30
1240	Arena	200	20

Después:

	cla	descripcion	pre	impue
	1200	Recubrimiento P1019	240	24
	1210	Recubrimiento P1028	250	25
	1220	Recubrimiento P1037	280	28
	1230	Cemento	300	30
	1240	Arena	200	20
	1250	Grava	100	10
	1260	Gravilla	90	9
	1270	Tezontle	80	8
	1280	Tepetate	34	3.4
	1290	Tubería 3.5	200	20
	1300	Tubería 4.3	210	21
	1310	Tubería 3.6	220	22
	1320	Tubería 4.4	230	23
	1330	Tubería 3.7	240	24
	1340	Tubería 4.5	250	25
	1350	Tubería 3.8	260	26
	1360	Pintura C1010	125	12.5
	1370	Pintura B1020	125	12.5
	1380	Pintura C1011	725	72.5
	1390	Pintura B1021	125	12.5
	1400	Pintura C1011	125	12.5
	1410	Pintura B1021	125	12.5
	1420	Pintura C1012	125	12.5
	1430	Pintura B1022	125	12.5
	2000	Jabón	125	12.5
	5000	Martillos Acme	250	15

**Crear procedimientos con los siguientes objetivos:**

**modificaMaterial** que permite modificar un material que reciba como parámetros las columnas de la tabla materiales y actualice las columnas correspondientes con los valores recibidos, para el registro cuya llave sea la clave que se recibe como parámetro.

```
DROP PROCEDURE IF EXISTS modificaMaterial;
DELIMITER //
CREATE PROCEDURE modificaMaterial(
    IN uclave INT(11),
    IN udescripcion VARCHAR(40),
    IN uprecio FLOAT,
    IN uimpuesto FLOAT
)
BEGIN
    UPDATE Materiales SET descripcion = udescripcion, precio = uprecio, impuesto = uimpuesto
    WHERE clave = uclave;
END //
```

```
CALL modificaMaterial(5000, 'Martillos FA', 300, 15);
```

Antes:

2000	Jabón	125	12.5
5000	Martillos Acme	250	15

Después:

2000	Jabón	125	12.5
5000	Martillos FA	300	15

**eliminaMaterial** que elimina el registro de la tabla materiales cuya llave sea la clave que se recibe como parámetro.

```
DROP PROCEDURE IF EXISTS eliminaMaterial;
DELIMITER //
CREATE PROCEDURE eliminaMaterial(
    IN uclave INT(11)
)
BEGIN
    DELETE FROM entregan WHERE clave = uclave;
    DELETE FROM materiales WHERE clave = uclave;
END //
```

```
CALL eliminaMaterial(5000);
```

Antes:

1420	Pintura C1012	125	12.5
1430	Pintura B1022	125	12.5
2000	Jabón	125	12.5
5000	Martillos FA	300	15

Después:

1420	Pintura C1012	125	12.5
1430	Pintura B1022	125	12.5
2000	Jabón	125	12.5

**Crear procedimientos con los siguientes crearProyecto, modificaProyecto y eliminaProyecto**

**crearProyecto:**

```
DROP PROCEDURE IF EXISTS creaProyecto;
DELIMITER //
CREATE PROCEDURE creaProyecto(
    IN numero INT(11),
    IN denominacion VARCHAR(40)
)
BEGIN
    INSERT INTO proyectos VALUES(numero, denominacion);
END //
```

Antes:

numero	denominacion
5000	Vamos Mexico
5001	Aztecon
5002	CIT Campeche
5003	Mexico sin ti no estamos completos
5004	Educando en Coahuila
5005	Infonavit Durango
5006	Reconstrucción del templo de Guadalupe
5007	Construcción de plaza Magnolias
5008	Televisa en acción
5009	Disco Atlantic
5010	Construcción de Hospital Infantil
5011	Remodelación de aulas del IPP
5012	Restauración de instalaciones del CEA
5013	Reparación de la plaza Sonora
5014	Remodelación de Soriana
5015	CIT Yucatan
5016	Ampliación de la carretera a la huasteca
5017	Reparación de la carretera del sol
5018	Tu cambio por la educación
5019	Queretaro limpio

Después:

numero	denominacion
5000	Vamos Mexico
5001	Aztecon
5002	CIT Campeche
5003	Mexico sin ti no estamos completos
5004	Educando en Coahuila
5005	Infonavit Durango
5006	Reconstrucción del templo de Guadalupe
5007	Construcción de plaza Magnolias
5008	Televisa en acción
5009	Disco Atlantic
5010	Construcción de Hospital Infantil
5011	Remodelación de aulas del IPP
5012	Restauración de instalaciones del CEA
5013	Reparación de la plaza Sonora
5014	Remodelación de Soriana
5015	CIT Yucatan
5016	Ampliación de la carretera a la huasteca
5017	Reparación de la carretera del sol
5018	Tu cambio por la educación
5019	Queretaro limpio
5020	RECON

### modificaProyecto:

```
DROP PROCEDURE IF EXISTS modificaProyecto;
DELIMITER //
CREATE PROCEDURE modificaProyecto(
    IN unumero INT(11),
    IN udenominacion VARCHAR(40)
)
BEGIN
    UPDATE proyectos SET denominacion = udenominacion WHERE numero = unumero
;
END //

CALL modificaProyecto(5020, 'Natdev WBS');
```

### Antes:

numero	denominacion
5000	Vamos Mexico
5001	Aztecon
5002	CIT Campeche
5003	Mexico sin ti no estamos completos
5004	Educando en Coahuila
5005	Infonavit Durango
5006	Reconstrucción del templo de Guadalupe
5007	Construcción de plaza Magnolias
5008	Televisa en acción
5009	Disco Atlantic
5010	Construcción de Hospital Infantil
5011	Remodelación de aulas del IPP
5012	Restauración de instalaciones del CEA
5013	Reparación de la plaza Sonora
5014	Remodelación de Soriana
5015	CIT Yucatan
5016	Ampliación de la carretera a la huasteca
5017	Reparación de la carretera del sol
5018	Tu cambio por la educación
5019	Queretaro limpio
5020	RECON

### Después:

numero	denominacion
5000	Vamos Mexico
5001	Aztecon
5002	CIT Campeche
5003	Mexico sin ti no estamos completos
5004	Educando en Coahuila
5005	Infonavit Durango
5006	Reconstrucción del templo de Guadalupe
5007	Construcción de plaza Magnolias
5008	Televisa en acción
5009	Disco Atlantic
5010	Construcción de Hospital Infantil
5011	Remodelación de aulas del IPP
5012	Restauración de instalaciones del CEA
5013	Reparación de la plaza Sonora
5014	Remodelación de Soriana
5015	CIT Yucatan
5016	Ampliación de la carretera a la huasteca
5017	Reparación de la carretera del sol
5018	Tu cambio por la educación
5019	Queretaro limpio
5020	Natdev WBS

### eliminaProyecto:

```
DROP PROCEDURE IF EXISTS eliminaProyecto;
DELIMITER //
CREATE PROCEDURE eliminaProyecto(
    IN unumero INT(11)
```

```

)
BEGIN
    DELETE FROM proyectos WHERE numero = unumero;
END //

CALL eliminaProyecto(5020);

```

Antes:

numero	denominacion
5000	Vamos Mexico
5001	Aztecon
5002	CIT Campeche
5003	Mexico sin ti no estamos completos
5004	Educando en Coahuila
5005	Infonavit Durango
5006	Reconstrucción del templo de Guadalupe
5007	Construcción de plaza Magnolias
5008	Televisa en acción
5009	Disco Atlantic
5010	Construcción de Hospital Infantil
5011	Remodelación de aulas del IPP
5012	Restauración de instalaciones del CEA
5013	Reparación de la plaza Sonora
5014	Remodelación de Soriana
5015	CIT Yucatan
5016	Ampliación de la carretera a la huasteca
5017	Reparación de la carretera del sol
5018	Tu cambio por la educación
5019	Queretaro limpio
5020	Natdev WBS

Después:

numero	denominacion
5000	Vamos Mexico
5001	Aztecon
5002	CIT Campeche
5003	Mexico sin ti no estamos completos
5004	Educando en Coahuila
5005	Infonavit Durango
5006	Reconstrucción del templo de Guadalupe
5007	Construcción de plaza Magnolias
5008	Televisa en acción
5009	Disco Atlantic
5010	Construcción de Hospital Infantil
5011	Remodelación de aulas del IPP
5012	Restauración de instalaciones del CEA
5013	Reparación de la plaza Sonora
5014	Remodelación de Soriana
5015	CIT Yucatan
5016	Ampliación de la carretera a la huasteca
5017	Reparación de la carretera del sol
5018	Tu cambio por la educación
5019	Queretaro limpio

**Crear procedimientos con los siguientes crearProveedor, modificaProveedor y eliminaProveedor**

**crearProveedor:**

```

DROP PROCEDURE IF EXISTS creaProveedor;
DELIMITER //
CREATE PROCEDURE creaProveedor(
    IN rfc VARCHAR(15),
    IN razonsocial VARCHAR(40)
)
BEGIN
    INSERT INTO proveedores VALUES (rfc, razonsocial);
END //

```

```
CALL creaProveedor('IIII800101','Peñips');
```

Antes:

rfc	razonsocial
AAAA800101	La fragua
BBBB800101	Oviedo
CCCC800101	La Ferre
DDDD800101	Cecoferre
EEEE800101	Alvin
FFFF800101	Comex
GGGG800101	Tabiguera del centro
HHHH800101	Tubasa

Después:

rfc	razonsocial
AAAA800101	La fragua
BBBB800101	Oviedo
CCCC800101	La Ferre
DDDD800101	Cecoferre
EEEE800101	Alvin
FFFF800101	Comex
GGGG800101	Tabiguera del centro
HHHH800101	Tubasa
IIII800101	Peñips

**modificaProveedor:**

```
DROP PROCEDURE IF EXISTS modificaProveedor;
DELIMITER //
CREATE PROCEDURE modificaProveedor(
    IN urfc VARCHAR(15),
    IN urazonsocial VARCHAR(40)
)
BEGIN
    UPDATE Proveedores SET razonsocial = urazonsocial
    WHERE rfc = urfc;
END //

CALL modificaProveedor('IIII800101', 'TabiPeñips');
```

Antes:

rfc	razonsocial
AAAA800101	La fragua
BBBB800101	Oviedo
CCCC800101	La Ferre
DDDD800101	Cecoferre
EEEE800101	Alvin
FFFF800101	Comex
GGGG800101	Tabiguera del centro
HHHH800101	Tubasa
IIII800101	Peñips

Después:

rfc	razonsocial
AAAA800101	La fragua
BBBB800101	Oviedo
CCCC800101	La Ferre
DDDD800101	Cecoferre
EEEE800101	Alvin
FFFF800101	Comex
GGGG800101	Tabiguera del centro
HHHH800101	Tubasa
IIII800101	TabiPeñips



### eliminaProveedor:

```
DROP PROCEDURE IF EXISTS eliminaProveedor;
DELIMITER //
CREATE PROCEDURE eliminaProveedor(
    IN urfc VARCHAR(15)
)
BEGIN
    DELETE FROM proveedores WHERE rfc = urfc;
END //

CALL eliminaProveedor('IIII800101');
```

#### Antes:

rfc	razonsocial
AAAA800101	La fragua
BBBB800101	Oviedo
CCCC800101	La Ferre
DDDD800101	Cecoferre
EEEE800101	Alvin
FFFF800101	Comex
GGGG800101	Tabiquera del centro
HHHH800101	Tubasa
IIII800101	TabiPeñips

#### Después:

rfc	razonsocial
AAAA800101	La fragua
BBBB800101	Oviedo
CCCC800101	La Ferre
DDDD800101	Cecoferre
EEEE800101	Alvin
FFFF800101	Comex
GGGG800101	Tabiquera del centro
HHHH800101	Tubasa

### Crear procedimientos con los siguientes crearEntrega, modificaEntrega y eliminaEntrega

#### crearEntrega:

```
DROP PROCEDURE IF EXISTS creaEntrega;
DELIMITER //
CREATE PROCEDURE creaEntrega(
    IN clave INT(11),
    IN rfc VARCHAR(15),
    IN numero INT(11),
    IN fecha DATE,
    IN cantidad INT(11)
)
BEGIN
    INSERT INTO entregan VALUES (clave, rfc, numero, fecha, cantidad);
END //

CALL creaEntrega(1430, 'HHHH800101', 5018, '2020-04-14', 150);
```

Antes:

	clæ	rfc	numi	fecha	cantic
1390	HHHH800101	5019	2000-11-10	107	
1400	AAAA800101	5000	1999-04-07	382	
1400	AAAA800101	5010	2005-04-21	116	
1410	BBBB800101	5001	2000-05-18	601	
1410	BBBB800101	5009	2003-07-13	467	
1420	CCCC800101	5002	2001-09-09	603	
1420	CCCC800101	5008	1999-05-05	278	
1430	DDDD800101	5003	2005-04-30	576	
1430	DDDD800101	5007	2000-03-02	13	

Después:

	clæ	rfc	numi	fecha	cantic
1390	HHHH800101	5019	2000-11-10	107	
1400	AAAA800101	5000	1999-04-07	382	
1400	AAAA800101	5010	2005-04-21	116	
1410	BBBB800101	5001	2000-05-18	601	
1410	BBBB800101	5009	2003-07-13	467	
1420	CCCC800101	5002	2001-09-09	603	
1420	CCCC800101	5008	1999-05-05	278	
1430	DDDD800101	5003	2005-04-30	576	
1430	DDDD800101	5007	2000-03-02	13	
1430	HHHH800101	5018	2020-04-14	150	

modificaEntrega:

```
DROP PROCEDURE IF EXISTS modificaEntrega;
DELIMITER //
CREATE PROCEDURE modificaEntrega(
    IN uclave INT(11),
    IN urfc VARCHAR(15),
    IN unumero INT(11),
    IN ufecha DATE,
    IN ucantidad INT(11)
)
BEGIN
    UPDATE Entregan SET fecha = ufecha, cantidad = ucantidad
    WHERE clave = uclave AND rfc = urfc AND numero = unumero;
END //

CALL modificaEntrega(1430, 'HHHH800101', 5018, '2021-04-14', 300);
```

Antes:

	clæ	rfc	numi	fecha	cantic
1390	HHHH800101	5019	2000-11-10	107	
1400	AAAA800101	5000	1999-04-07	382	
1400	AAAA800101	5010	2005-04-21	116	
1410	BBBB800101	5001	2000-05-18	601	
1410	BBBB800101	5009	2003-07-13	467	
1420	CCCC800101	5002	2001-09-09	603	
1420	CCCC800101	5008	1999-05-05	278	
1430	DDDD800101	5003	2005-04-30	576	
1430	DDDD800101	5007	2000-03-02	13	
1430	HHHH800101	5018	2020-04-14	150	

Después:

	clæ	rfc	numi	fecha	cantic
	HHHH800101	5019	2000-11-10	107	
	AAAA800101	5000	1999-04-07	382	
	AAAA800101	5010	2005-04-21	116	
	BBBB800101	5001	2000-05-18	601	
	BBBB800101	5009	2003-07-13	467	
	CCCC800101	5002	2001-09-09	603	
	CCCC800101	5008	1999-05-05	278	
	DDDD800101	5003	2005-04-30	576	
	DDDD800101	5007	2000-03-02	13	
	HHHH800101	5018	2021-04-14	300	

**eliminaEntrega:**

```
DROP PROCEDURE IF EXISTS eliminaEntrega;
DELIMITER //
CREATE PROCEDURE eliminaEntrega(
    IN uclave INT(11),
    IN urfc VARCHAR(15),
    IN unumero INT(11)
)
BEGIN
    DELETE FROM entregan WHERE clave = uclave AND rfc = urfc AND numero = unumero;
END //

CALL eliminaEntrega(1430, 'HHHH800101', 5018);
```

**Antes:**

clz	rfc	numi	fecha	cantic
HHHH800101	5019	2000-11-10	107	
AAAA800101	5000	1999-04-07	382	
AAAA800101	5010	2005-04-21	116	
BBBB800101	5001	2000-05-18	601	
BBBB800101	5009	2003-07-13	467	
CCCC800101	5002	2001-09-09	603	
CCCC800101	5008	1999-05-05	278	
DDDD800101	5003	2005-04-30	576	
DDDD800101	5007	2000-03-02	13	
HHHH800101	5018	2021-04-14	300	

**Después:**

	clz	rfc	numi	fecha	cantic
1390	HHHH800101	5011	2003-02-16	697	
1390	HHHH800101	5019	2000-11-10	107	
1400	AAAA800101	5000	1999-04-07	382	
1400	AAAA800101	5010	2005-04-21	116	
1410	BBBB800101	5001	2000-05-18	601	
1410	BBBB800101	5009	2003-07-13	467	
1420	CCCC800101	5002	2001-09-09	603	
1420	CCCC800101	5008	1999-05-05	278	
1430	DDDD800101	5003	2005-04-30	576	
1430	DDDD800101	5007	2000-03-02	13	

**Define el siguiente stored procedure en tu base de datos:**

```
IF EXISTS (SELECT name FROM sysobjects
            WHERE name = 'queryMaterial' AND type = 'P')
    DROP PROCEDURE queryMaterial
GO

CREATE PROCEDURE queryMaterial
    @udescription VARCHAR(50),
    @ucosto NUMERIC(8,2)
AS
    SELECT * FROM Materiales WHERE descripcion
    LIKE '%' + @udescription + '%' AND costo > @ucosto
GO
```

Ejecútalo con la siguiente instrucción:

```
EXECUTE queryMaterial 'Lad',20
```

**Explica en tu reporte qué recibe como parámetro este procedimiento y qué hace.**

Este procedimiento recibe como parámetros la descripción de un Material y un costo del material; lo que se realiza en el procedimiento es buscar materiales cuya descripción contenga la cadena ingresada en descripción y que tenga un costo mayor al recibido.

**¿Qué ventajas tienen el utilizar Stored Procedures en una aplicación cliente-servidor?**

- Aumentan escalabilidad dado que se aísla el procesamiento de datos de la aplicación, haciéndolo tan solo en el servidor.
- El mantenimiento de un SP es más fácil dado que solo se cambia una vez en el servidor en lugar de en todas las copias de clientes.

**¿Qué ventajas tiene utilizar SP en un proyecto?**

- Son más rápidos y eficientes dado que solo se tienen que compilar una vez y se almacenan sus ejecutables haciendo las consultas más rápidas.
- Reutilización de código, se logra mayor productividad ya que se usa una y otra vez el código.