Martín Helmut Domínguez Álvarez A01701813

Lab 21.

```
IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'creaMaterial' AND type = 'P')

DROP PROCEDURE creaMaterial

GO

CREATE PROCEDURE creaMaterial

@uclave NUMERIC(5,0),

@udescripcion VARCHAR(50),

@ucosto NUMERIC(8,2),

@uimpuesto NUMERIC(6,2)

AS

INSERT INTO Materiales VALUES(@uclave, @udescripcion, @ucosto, @uimpuesto)

GO
```

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

Si ya existe el procedure crearMaterial se hace un drop procedure y se crea de nuevo, de lo contrario sólo se crea.

¿Para qué sirve la instrucción GO?

GO es un separador de batch (bloque), no temina la instrucción.

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

Los parámetros que recibe son las columnas clave, descripsión, costo e impuesto que modifican la tabla Materiales.

```
EXECUTE creaMaterial 5000, 'Martillos Acme', 250, 15

SELECT *FROM Materiales
```

Por analogía crea procedimientos almacenados 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.

```
    IF EXISTS (SELECT name FROM sysobjects
        WHERE name = 'modificaMaterial' AND type = 'P')
        DROP PROCEDURE modificaMaterial
        GO
        create procedure modificaMaterial
            @uclave NUMERIC(5,0),
            @udescripcion VARCHAR(50),
            @ucosto NUMERIC(8,2),
            @uimpuesto NUMERIC(6,2)
        AS
            UPDATE Materiales SET
            Clave=@uclave, Descripcion=@udescripcion, Costo=@ucosto, PorcentajeImpuesto=@uimpuesto
            WHERE Clave=@uclave
```

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

```
IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'eliminaMaterial' AND type = 'P')

DROP PROCEDURE eliminaMaterial

GO

create procedure eliminaMaterial

@uclave NUMERIC(5,0),

@udescripcion VARCHAR(50),

@ucosto NUMERIC(8,2),

@uimpuesto NUMERIC(6,2)

AS

DELETE from Materiales

WHERE Clave=@uclave

GO
```

• Desarrollar los procedimientos (almacenados) creaProyecto, modificaproyecto y eliminaproyecto, hacer lo mismo para las tablas proveedores y entregan.

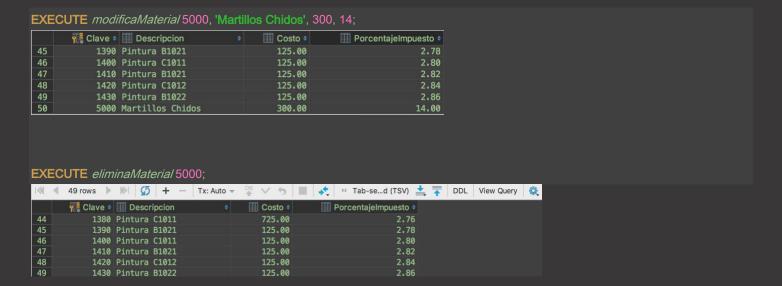
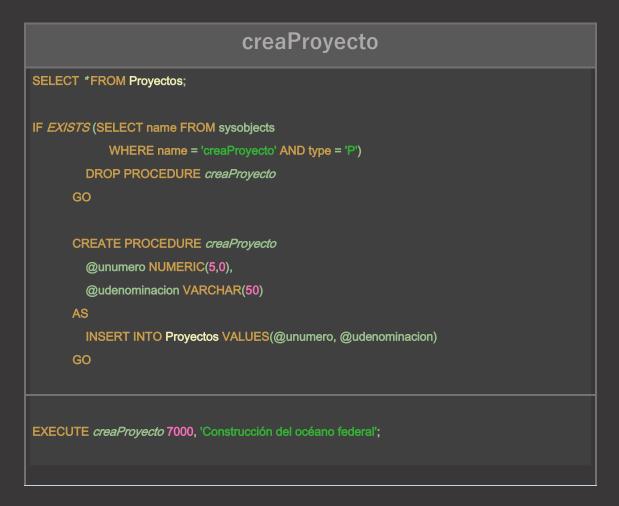
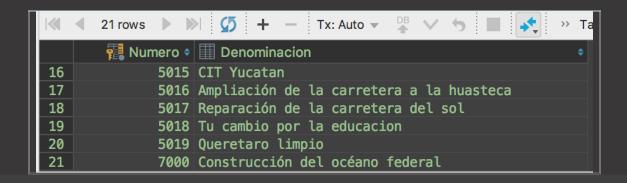


Tabla Proyecto







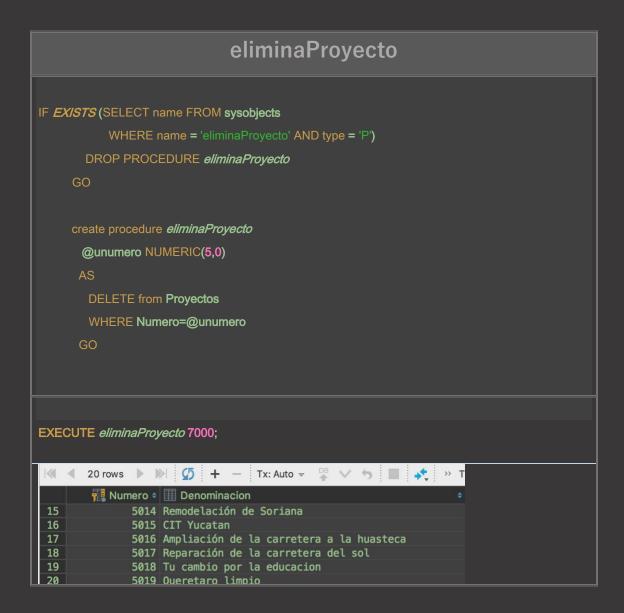


Tabla Proveedor

```
IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'creaProveedor' AND type = 'P')

DROP PROCEDURE creaProveedor

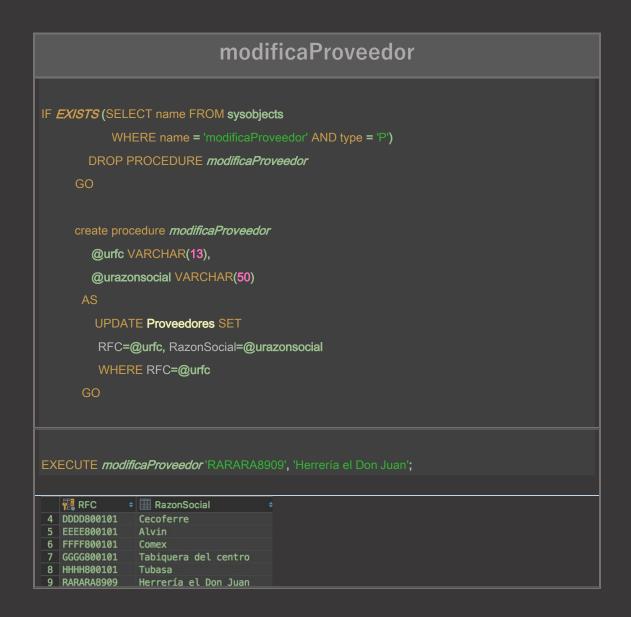
GO

CREATE PROCEDURE creaProveedor

@urfc VARCHAR(13),

@urazonsocial VARCHAR(50)
```





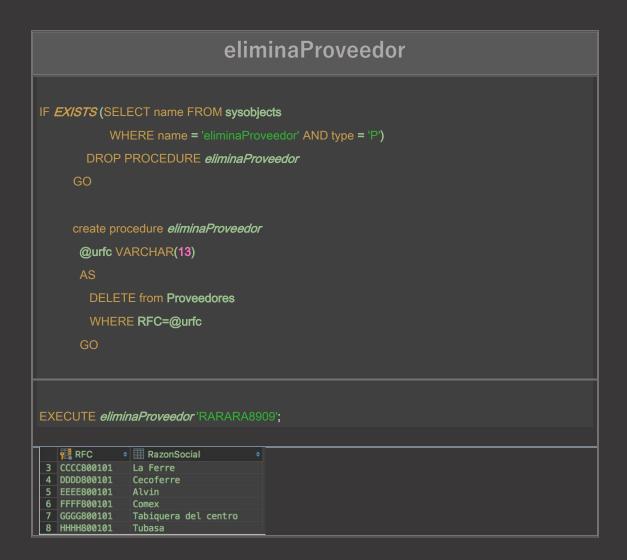


Tabla Entrega

```
IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'creaEntrega' AND type = 'P')

DROP PROCEDURE creaEntrega

GO

CREATE PROCEDURE creaEntrega

@uclave NUMERIC(5),
```

```
@urfc VARCHAR(13),
@unumero NUMERIC(5,0),
@ufecha datetime,
@ucantidad NUMERIC(8,2)
AS
INSERT INTO Entregan VALUES(@uclave, @urfc, @unumero, @ufecha,@ucantidad)
GO
```

EXECUTE creaEntrega 5000, 'RARARA8909', 7000, '2000/04/09', 234;

	Clave 🕏	RFC RFC	📆 Numero 🕏	Fecha	*	Ⅲ Cantidad 🕏
128	1420	CCCC800101	5008	2000-08-02 00:00:00.000		278.00
129	1420	CCCC800101	5008	2001-02-12 00:00:00.000		444.00
130	1430	DDDD800101	5003	1999-09-02 00:00:00.000		576.00
131	1430	DDDD800101	5007	1998-01-09 00:00:00.000		13.00
132	1430	DDDD800101	5007	2002-06-10 00:00:00.000		506.00
133	5000	RARARA8909	7000	2000-04-09 00:00:00.000		234.00

modificaEntrega

```
IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'modificaEntrega' AND type = 'P')

DROP PROCEDURE modificaEntrega

GO

create procedure modificaEntrega

@uclave NUMERIC(5),
@urfc VARCHAR(13),
@unumero NUMERIC(5,0),
@ufecha datetime,
@ucantidad NUMERIC(8,2)

AS

UPDATE Entregan SET

RFC=@urfc, Clave=@uclave, Cantidad=@ucantidad, Fecha=@ufecha, Numero=@unumero
WHERE Clave=@uclave AND RFC=@urfc and Numero=@unumero and Fecha=@ufecha
GO
```

EXECUTE *modificaEntrega* **5000**, 'RARARA8909', **7000**, '2000/04/09', **600**;

	📆 Clave 🕏	RFC ÷	Numero 🕈	Fecha	\$	Cantidad +
128	1420	CCCC800101	5008	2000-08-02 00:00:00.000		278.00
129	1420	CCCC800101	5008	2001-02-12 00:00:00.000		444.00
130	1430	DDDD800101	5003	1999-09-02 00:00:00.000		576.00
131	1430	DDDD800101	5007	1998-01-09 00:00:00.000		13.00
132	1430	DDDD800101	5007	2002-06-10 00:00:00.000		506.00
133	5000	RARARA8909	7000	2000-04-09 00:00:00.000		600.00

eliminaEntrega

```
IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'eliminaEntrega' AND type = 'P')

DROP PROCEDURE eliminaEntrega

GO

create procedure eliminaEntrega

@uclave NUMERIC(5),

@urfc VARCHAR(13),

@unumero NUMERIC(5,0),

@ufecha datetime

AS

DELETE from Entregan

WHERE Clave=@uclave AND RFC=@urfc and Numero=@unumero and Fecha=@ufecha
```

EXECUTE *eliminaEntrega* 5000, 'RARARA8909', 7000, '2000/04/09';

	🔁 Clave 🕏	RFC	\$	📆 Numero 🕏	📆 Fecha		\$	Cantidad +
127	1420	CCCC800101		5002	1998-04-07	00:00:00.000		603.00
128	1420	CCCC800101		5008	2000-08-02	00:00:00.000		278.00
129	1420	CCCC800101		5008	2001-02-12	00:00:00.000		444.00
130	1430	DDDD800101		5003	1999-09-02	00:00:00.000		576.00
131	1430	DDDD800101		5007	1998-01-09	00:00:00.000		13.00
132	1430	DDDD800101		5007	2002-06-10	00:00:00.000		506.00

```
Explica en tu reporte qué recibe como parámetro este Procedimiento y qué hace

IF EXISTS (SELECT name FROM sysobjects

WHERE name = 'queryMaterial' AND type = 'P')

DROP PROCEDURE queryMaterial
```

CREATE PROCEDURE queryMaterial

@udescripcion VARCHAR(50),

@ucosto NUMERIC(8,2)

AS

SELECT *FROM Materiales WHERE descripcion

LIKE '%'+@udescripcion+'%' AND costo > @ucosto

GO

EXECUTE queryMaterial 'Lad',20;

	Clave +	Descripcion	†	Costo ÷	Porcentajelmpuesto 🕏
1	1080	Ladrillos	rojos	50.00	2.16
2	1090	Ladrillos	grises	35.00	2.18

Este prcedimiento recibe un string y un costo, lo que hace el procedimiento es que busca los materiales que tienen ese fragmento de palabras y que tiene un costo mayor al valor que se metió como parámetro

Ejecutar Stored Procedures desde la aplicación cliente
Para uno de tus laboratorios anteriores o proyecto, crea un stored procedure en MySQL de
manera similar a cómo lo acabas de hacer en SQL Server. Posteriormente agrega a tu
librería de la capa del modelo, una función de php para ejecutar tu stored procedure y
prueba su funcionamiento desde la aplicación. Toma como referencia la guía de
procedimientos almacenados de php.

DELIMITER //

CREATE PROCEDURE addProduct

(IN uid INT(20), IN unombre VARCHAR(11), IN udescripcion TEXT, IN uatributos VARCHAR(11))

BEGIN

INSERT INTO product(id, nombre, descripcion, atributos) VALUES (uid,unombre, udescripcion, uatributos);

FND //

DELIMITER;

Funciona

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

Simplifica mucho el manejo de funciones o consultas recurrentes. Te evita el reescribir mucho código.

¿Qué ventajas tiene utilizar SP en un proyecto?

Permite usar llamadas genéricas para invocar funciones ya definidas, esto acorta mucho el trabajo y permite una mayor legibilidad, además de que vuelve hace más fácil el manejo de las bases de datos. No tienes que cambiar todas las funciones para corregir algo, basta con hacerlo en el procedure.

Para reforzar el entendimiento de los SP, crea un SP que sea de utilidad en tu proyecto (en caso de usar otra tecnología, se deberá asumir). Incorpora dicho SP en tu reporte.

DELIMITER //

CREATE PROCEDURE creaUsuario

(IN uid_usuario VARCHAR(12), IN unombre TINYTEXT, IN uemail TINYTEXT, IN udireccion TINYTEXT, IN utelefono VARCHAR(10))

REGIN

INSERT INTO product(id_usuario, nombre, email, dirección, telefono) VALUES (uid_usuario ,unombre, uemail , udireccion, utelefono);

END //

DELIMITER;

Funciona