**Michelle Fernanda Anguiano Juárez – A01209529**

**Lab 21**

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

Verifica si el procedimiento ya existe, y en caso de existir, lo elimina.

* **¿Para qué sirve la instrucción GO?**

Es una instrucción para ejecutar las sentencias anteriores.

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

Recibe como parámetros la clave, descripción, costo e impuesto, y modifica la tabla 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 Descripcion**=@udescripcion, **Costo**=@ucosto, **PorcentajeImpuesto**=@uimpuesto  
 **WHERE Clave**=@uclave  
**GO**

**EXECUTE** *modificaMaterial* 5000,**'Martillos Acme'**,250,20  
  
**SELECT** *\** **FROM** Materiales

* **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)  
**AS  
 DELETE FROM** Materiales  
 **WHERE** Clave=@uclave  
**GO  
  
EXECUTE** *eliminaMaterial* 5000  
  
**SELECT** *\** **FROM** Materiales

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

* **creaProyecto**

**IF** *EXISTS* (**SELECT name FROM** sysobjects  
 **WHERE name** = **'creaProyecto' AND type** = **'P'**)  
 **DROP PROCEDURE** *creaProyecto***GO  
  
CREATE PROCEDURE** *creaProyecto* @unumero **NUMERIC**(5,0),  
 @udenominacion **VARCHAR**(50)  
**AS  
 INSERT INTO** Proyectos **VALUES**(@unumero, @udenominacion)  
**GO**

* **modificaProyecto**

**IF** *EXISTS* (**SELECT name FROM** sysobjects  
 **WHERE name** = **'modificaProyecto' AND type** = **'P'**)  
 **DROP PROCEDURE** *modificaProyecto***GO  
  
CREATE PROCEDURE** *modificaProyecto* @unumero **NUMERIC**(5,0),  
 @udenominacion **VARCHAR**(50)  
**AS  
 UPDATE** Proyectos  
 **SET Denominacion**=@udenominacion  
 **WHERE Numero**=@unumero  
**GO**

* **eliminaProyecto**

**IF** *EXISTS* (**SELECT name FROM** sysobjects  
 **WHERE name** = **'eliminaProyecto' AND type** = **'P'**)  
 **DROP PROCEDURE** *eliminaProyecto***GO  
  
CREATE PROCEDURE** *eliminaProyecto* @unumero **NUMERIC**(5,0)  
**AS  
 DELETE FROM** Proyectos  
 **WHERE** Numero=@unumero  
**GO**

**Crear procedimientos para realizar consultas con parámetros**

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

* **¿Qué recibe como parámetro este procedimiento y qué hace?**

Recibe como parámetros la descripción y el costo de un material. Lo que hace es ejecutar una consulta que devuelve los datos de los materiales cuya descripción contenga la cadena recibida como parámetro, y cuyo costo sea mayor al que se especifica.

* **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**](http://php.net/manual/es/mysqli.quickstart.stored-procedures.php)**.**

*-- STORED PROCEDURES PROVEEDORES  
  
-- creaProveedor***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'creaProveedor' AND type** = **'P'**)  
 **DROP PROCEDURE** *creaProveedor***GO  
   
CREATE PROCEDURE** *creaProveedor* @urfc **CHAR**(13),  
 @urazonsocial **VARCHAR**(50)  
**AS  
 INSERT INTO** Proveedores **VALUES**(@urfc, @urazonsocial)  
**GO***-- modificaProveedor***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'modificaProveedor' AND type** = **'P'**)  
 **DROP PROCEDURE** *modificaProveedor***GO  
   
CREATE PROCEDURE** *modificaProveedor* @urfc **CHAR**(13),  
 @urazonsocial **VARCHAR**(50)  
**AS  
 UPDATE** Proveedores   
 **SET RazonSocial**=@urazonsocial  
 **WHERE RFC**=@urfc  
**GO***-- eliminaProveedor***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'eliminaProveedor' AND type** = **'P'**)  
 **DROP PROCEDURE** *eliminaProveedor***GO  
   
CREATE PROCEDURE** *eliminaProveedor* @urfc **CHAR**(13)  
**AS  
 DELETE FROM** Proveedores   
 **WHERE** RFC=@urfc  
**GO***-- STORED PROCEDURES ENTREGAN  
  
-- creaEntrega***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'creaEntrega' AND type** = **'P'**)  
 **DROP PROCEDURE** *creaEntrega***GO  
   
CREATE PROCEDURE** *creaEntrega* @uclave **NUMERIC**(5,0),  
 @urfc **CHAR**(13),  
 @unumero **NUMERIC**(5,0),  
 @ufecha **DATETIME**,  
 @ucantidad **NUMERIC**(8,2)  
**AS  
 INSERT INTO** Entregan **VALUES**(@uclave, @urfc, @unumero, @ufecha, @ucantidad)  
**GO  
  
SELECT** *\** **FROM** Entregan  
  
**SET DATEFORMAT** DMY  
**EXECUTE** *creaEntrega* 1000,**'AAAA800101'**,5000,**'15/06/2017'**,500  
  
*-- modificaEntrega***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'modificaEntrega' AND type** = **'P'**)  
 **DROP PROCEDURE** *modificaEntrega***GO  
   
CREATE PROCEDURE** *modificaEntrega* @uclave **NUMERIC**(5,0),  
 @urfc **CHAR**(13),  
 @unumero **NUMERIC**(5,0),  
 @ufecha **DATETIME**,  
 @ucantidad **NUMERIC**(8,2)  
**AS  
 UPDATE** Entregan   
 **SET Cantidad**=@ucantidad  
 **WHERE Clave**=@uclave  
 **AND RFC**=@urfc  
 **AND Numero**=@unumero  
 **AND Fecha**=@ufecha  
**GO  
  
SELECT** *\** **FROM** Entregan  
  
**SET DATEFORMAT** DMY  
**EXECUTE** *modificaEntrega* 1000,**'AAAA800101'**,5000,**'15/06/2017'**,600  
  
*-- eliminaEntrega***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'eliminaEntrega' AND type** = **'P'**)  
 **DROP PROCEDURE** *eliminaEntrega***GO  
   
CREATE PROCEDURE** *eliminaEntrega* @uclave **NUMERIC**(5,0),  
 @urfc **CHAR**(13),  
 @unumero **NUMERIC**(5,0),  
 @ufecha **DATETIME  
AS  
 DELETE FROM** Entregan   
 **WHERE** Clave=@uclave  
 **AND** RFC=@urfc  
 **AND** Numero=@unumero  
 **AND** Fecha=@ufecha  
**GO  
  
SELECT** *\** **FROM** Entregan  
  
**SET DATEFORMAT** DMY  
**EXECUTE** *eliminaEntrega* 1000,**'AAAA800101'**,5000,**'15/06/2017'***-- Crear procedimientos para realizar consultas con parámetros***IF** *EXISTS* (**SELECT name FROM** sysobjects   
 **WHERE name** = **'queryMaterial' AND type** = **'P'**)  
 **DROP PROCEDURE** *queryMaterial***GO  
   
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

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

Mejora el rendimiento cuando se ejecutan sentencias SQL, porque la aplicación cliente pasa el control a un Stored Procedure en el servidor. El procedimiento realiza un procesamiento intermedio, sin transmitir datos innecesarios en la red.

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

Los SP son una manera de tener un código más estructurado, simplificando su administración y mantenimiento. Además, permiten ejecutar sentencias SQL desde funciones en PHP sin necesidad de conocer toda la sintaxis de SQL.

DROP PROCEDURE `capturaGasto`;

CREATE DEFINER=`root`@`localhost` PROCEDURE ` capturaGasto`(IN `u\_idTipoEVento` INT(11), IN

`u\_tituloEvento` VARCHAR(40), IN `u\_descripcionEvento` TEXT, IN `u\_lugarEvento`

VARCHAR(140), IN `u\_fechaInicio` DATE, IN `u\_horaInicio` TIME, IN `u\_fechaFin` DATE, IN

`u\_horaFin` TIME) NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER

BEGIN

INSERT INTO evento (idTipoEvento, tituloEvento, descripcionEvento, lugarEvento, fechaInicio,

horaInicio, fechaFin, horaFin)

VALUES (u\_idTipoEVento, u\_tituloEvento, u\_descripcionEvento,u\_lugarEvento, u\_fechaInicio, u\_

horaInicio, u\_fechaFin, u\_horaFin);

END