



INTRODUCCIÓN A LOS SISTEMAS DE BASES DE DATOS

Laboratorio 7 - Sección 3

Luciano Ignacio Revillod Jerez

La base de datos utilizadas tiene el nombre “zapatillasdb”.

Los procedimientos almacenados de la base de datos son:

- **InsertZapatilla:** Este procedimiento almacenado tiene la función de insertar un producto (zapatilla) en la tabla “zapatillas”.

Estructura:

```
CREATE PROCEDURE InsertZapatilla  
(ProductID INT, Modelo VARCHAR(20), Stock INT, Material VARCHAR(20),  
Precio INT, Talla VARCHAR(15))  
INSERT INTO zapatillas(ProductID, Modelo, Stock, Material, Precio, Talla)  
VALUES (ProductID, Modelo, Stock, Material, Precio, Talla)
```

- **UpdatePrecio:** Este procedimiento almacenado tiene la función de actualizar el precio de un producto en la tabla “zapatillas”, esto mediante su ID (ProductID) y un número correspondiente al porcentaje.

Estructura:

```
CREATE PROCEDURE UpdatePrecio (IDproducto INT, porcentaje INT)  
UPDATE zapatillas  
SET Precio = Precio + (Precio*porcentaje)/100  
WHERE ProductID = IDproducto
```

- **SelectAll:** Este procedimiento almacenado se encarga de mostrar toda la información de la tabla zapatillas, esto mediante una consulta SELECT *.

Estructura:

```
CREATE PROCEDURE SelectAll()  
SELECT * FROM Zapatillas
```

Los Triggers (Disparadores) de la base de datos son:

Insert_Zapatilla_T: El siguiente Trigger cumple la función de detectar cuando se realiza una sentencia “INSERT” y después mediante “AFTER” recolecta una serie de datos programados.

Estructura:

```
CREATE TRIGGER Insert_Zapatilla_T  
AFTER INSERT ON zapatillas  
FOR EACH ROW  
INSERT INTO temp_zapatillas_Insert (logID, ProductID, Stock, Fecha_Hora)  
VALUES (logID,new.ProductID,new.Stock, now())
```

Update_Precio_T: Este Trigger se encarga de detectar cuando se realiza una sentencia UPDATE, luego mediante “old.” y “new.”, selecciona los datos antiguos y nuevos del precio actualizado.

Estructura:

```
CREATE TRIGGER Update_Precio_T  
BEFORE UPDATE ON zapatillas  
FOR EACH ROW  
INSERT INTO temp_precio_update(logID, ProductID, OldPrecio, NewPrecio,  
Fecha_Hora)  
VALUES (logID, new.ProductID, old.Precio, new.Precio, now())
```