



**TECNOLÓGICO  
NACIONAL DE MÉXICO®**



**Ing. En Sistemas Computacionales**

**Taller de Base de Datos**

**Maestro:**

MC Daniel Esparza Soto

**Unidad 4 - Tarea:**

“Triggers”

**Alumnos:**

Grande Espinoza, Víctor Ramon

20170684

Lie Norzagaray, Luis

20170717

Reza Arzola, Luis Orlando

20170796

**04 / 05 / 2023**

1.- Utilizando trigger, validar que solo se vendan ordenes de lunes a viernes.

```
CREATE TRIGGER TR_OrdenesLunesViernes
ON Orders FOR INSERT AS
DECLARE @Dia INT

SELECT @Dia = DATEPART(dw, OrderDate) FROM inserted

IF @Dia IN(1, 7)
BEGIN
    ROLLBACK TRAN
    RAISERROR('Solo se puede vender los dias Lunes a Viernes.', 16, 1)
END
GO
```

2.- Validar que no se vendan mas de 20 ordenes por empleado en una semana.

```
CREATE TRIGGER TR_MaxOrdenesSemanal
ON Orders FOR INSERT
AS
BEGIN
    DECLARE @EmpID INT, @Fecha DATETIME, @Semana DATETIME, @Ordenes INT

    SELECT @EmpID = EmployeeID FROM inserted
    SELECT @Fecha = OrderDate FROM inserted

    SELECT @Semana = DATEPART(WK, @Fecha)
    SELECT @Ordenes = COUNT(*) FROM Orders
    WHERE EmployeeID = @EmpID AND DATEPART(WK, OrderDate) = @Semana

    IF (@Ordenes > 20)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('Solo puede vender 20 articulos por empleado.', 16, 1)
    END
END
GO
```

3.- Validar que el campo firstname en la tabla employees solamente tenga nombres que inicien con vocal.

```
CREATE TRIGGER TR_EmpleadoVocal
ON Employees FOR INSERT
AS
BEGIN
    DECLARE @Nombre NVARCHAR(50)

    SELECT @Nombre = FirstName FROM inserted

    IF (@Nombre = '[aeiou]%' )
    BEGIN
        ROLLBACK TRAN
        RAISERROR('Solamete se puede insertar nombres que inicien con un
vocal.', 16, 1)
    END
END
GO
```

4.- validar que el importe de venta de cada orden no sea mayor a \$10,000.

```
CREATE TRIGGER TR_ImporteMax
ON [Order Details] FOR INSERT
AS
BEGIN
    DECLARE @Importe INT

    SELECT @Importe = Quantity * UnitPrice FROM inserted

    IF (@Importe > 10000)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('El importe total de venta no debe ser mayor de $10,000.',
16, 1)
    END
END
GO
```

5.- validar que no se puedan eliminar ordenes que se hicieron los lunes.

```
CREATE TRIGGER TR_NoEliminarLunes
ON Orders FOR DELETE
AS
BEGIN
    DECLARE @Fecha DATETIME

    SELECT @Fecha = DATEPART(DW, OrderDate) FROM deleted

    IF (@Fecha = 2)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('No se pueden eliminar ordenes de los dias Lunes.', 16, 1)
    END
END
GO
```

6.- Validar que no se realicen inserciones masivas en la tabla products.

```
CREATE TRIGGER TR_NoInsertMasivasProducts
ON Products FOR INSERT
AS
BEGIN
    DECLARE @Contador INT

    SELECT @Contador = COUNT(*) FROM inserted

    IF (@Contador > 1)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('No se puede realizar inserciones masivas en esta tabla.',
16, 1)
    END
END
GO
```

7.- Validar que no se pueda modificar el campo unitprice de la tabla [order details].

```
CREATE TRIGGER TR_NoUpdateUnitPrice
ON [Order Details] FOR UPDATE
AS
BEGIN
    IF UPDATE(UnitPrice)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('No se pueden actualizar el campo UnitPrice de esta tabla.',
16, 1)
    END
END
GO
```

8.- Validar que solo se pueda actualizar una sola vez el nombre del cliente.

```
--      Agregar Contador a tabla Customers
ALTER TABLE Customers ADD Contador INT
GO
UPDATE Customers SET Contador = 0
GO
ALTER TABLE Customers ADD CONSTRAINT DF_Contador DEFAULT (0) FOR Contador
GO

--      Creacion del trigger
CREATE TRIGGER TR_ActualizaNombreUnaVez
ON Customers FOR UPDATE
AS
BEGIN
    DECLARE @Clave INT, @Conta INT

    SELECT @Clave = CustomerID, @Conta = ISNULL(Contador, 0) FROM inserted

    IF UPDATE(ContactName)
    BEGIN
        IF (@Conta > 1)
        BEGIN
            ROLLBACK TRAN
            RAISERROR('Solo se puede actualizar una sola vez el nombre del
cliente.', 16, 1)
        END
        ELSE
        BEGIN
            Update Customers SET Contador = @Conta + 1 WHERE CustomerID =
@Clave
        END
    END
END
GO
```

9.- Validar que no se puedan eliminar categorías que tengan una clave impar.

```
CREATE TRIGGER TR_NoEliminarClaveImpar
ON Categories FOR DELETE
AS
BEGIN
    DECLARE @CatID INT

    SELECT @CatID = CategoryID FROM deleted

    IF (@CatID % 2 = 0)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('No se puede eliminar categorias que tengan una clave
impar.', 16, 1)
    END
END
GO
```

10.- Validar que no se puedan insertar ordenes que se realicen en domingo.

```
CREATE TRIGGER TR_NoInsertarOrdenesDomingos
ON Orders FOR INSERT
AS
BEGIN
    DECLARE @Dia INT

    SELECT @DIA = DATEPART(DW, OrderDate) FROM inserted

    IF (@Dia = 1)
    BEGIN
        ROLLBACK TRAN
        RAISERROR('No se puede insertar ordenes los domingos', 16, 1)
    END
END
GO
```