

**TRIGGERS**

# Triggers

A trigger is a database object that is attached to a table. Triggers are often referred to as a "special kind of stored procedure". The main difference between a trigger and a stored procedure is that the trigger is attached to a table and is only fired when an INSERT, UPDATE or DELETE occurs. You specify the modification action(s) that fire the trigger when it is created.

- **Syntax**

```
CREATE TRIGGER trigger_EID
ON table_EID
FOR INSERT|UPDATE |DELETE
AS
BEGIN
    SQL Statements;
END;
```

# Triggers

Example 1: Trigger to update the stock when product is sold.

```
CREATE TRIGGER TR_INVENT_UPDATE
ON SALES
FOR INSERT
AS
BEGIN
UPDATE INVENT SET StockQty = StockQty- (SELECT QTY FROM INSERTED )
WHERE PID = (SELECT PID FROM INSERTED);
END;
```

# Triggers

Example 2: Trigger to delete the order if the product is deleted from the inventory.

```
CREATE TRIGGER TR_SALE_DELETE
ON INVENT
FOR DELETE
AS
BEGIN
DELETE FROM SALES WHERE PID = (SELECT PID FROM DELETED);
END;
```

# Triggers

Example 3: Trigger to update the stock when the order quantity has been updated.

```
CREATE TRIGGER TR_STOCK_UPDATE2
ON SALES
FOR UPDATE
AS
BEGIN
    UPDATE Stock SET SQty = SQty + (SELECT QTY FROM DELETED)
    WHERE PID = (SELECT PID FROM DELETED);

    UPDATE Stock SET SQty = SQty - (SELECT QTY FROM INSERTED)
    WHERE PID = (SELECT PID FROM INSERTED);

END;
```



# Triggers

Example 4: Trigger to check & update the stock when the order is placed

```
CREATE TRIGGER TR_INVENT_CHECK
ON SALES
FOR INSERT
AS
BEGIN
    DECLARE @QS AS INT;
    DECLARE @QR AS INT;
    SET @QR= ( SELECT QTY FROM INSERTED);
    SET @QS = (SELECT StockQty FROM INVENT WHERE PID=(SELECT PID FROM inserted));
    IF @QS >= @QR
        Begin
            UPDATE INVENT SET StockQty = StockQty- (SELECT QTY FROM INSERTED )
            WHERE PID = (SELECT PID FROM INSERTED);
            COMMIT;
        end
    ELSE
        ROLLBACK;
END;
```



