

## Support Bulletin # 107 How to create a simple update trigger

### Table of Contents

Introduction.....	1
Step 1 – Create the Audit Table.....	1
Step 2 – Create the Trigger.....	1
View/Delete the trigger.....	2
View Audit Table.....	2

### ***Introduction***

This document outlines how to create an update SQL Trigger will capture a changes to the products.taxcategory field. An Audit table will hold the old value, new value, system user name, product code, date and time of update.

You could substitute the products.taxcategory field for any field on the table. You need to make sure your Audit table columns are defined correctly and match the corresponding products table fields.

Another modification could be to change the product table to another table such as the customer table and test one of the customer fields.

### ***Step 1 – Create the Audit Table***

Run this script to create the audit table.

```
create table Audit (NewValue varchar(25), Oldvalue varchar(25), WindowsUser  
varchar(128), ProductCode char(25), UpdateDate datetime)
```

### ***Step 2 – Create the Trigger***

Run this script to create the trigger. The trigger will write values to the Audit table if the products.taxcategory field is update.

It's important that the fields in the DECLARE section match the field type and length in the products table and audit table.

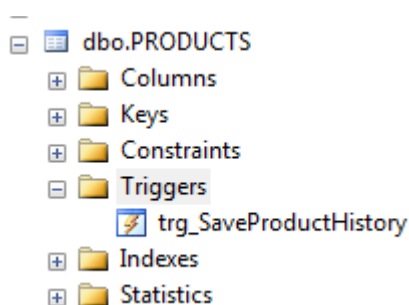
```
CREATE TRIGGER trg_SaveProductHistory ON Products
FOR INSERT, UPDATE
AS
IF UPDATE (TAXCATEGORY)
BEGIN
    DECLARE @newValue varchar(25)
    DECLARE @oldValue varchar(25)
    DECLARE @itemId varchar(25)
    DECLARE @UpdateDate varchar(21)
    DECLARE @UserName varchar(128)

    select @UserName = system_user,
    @UpdateDate = convert(varchar(8), getdate(), 112) + ' ' +
convert(varchar(12), getdate(), 114)

    SET @newValue = (SELECT TAXCATEGORY FROM Inserted)
    SET @oldValue = (SELECT TAXCATEGORY FROM Deleted)
    IF @newValue != @oldValue
    BEGIN
        SET @itemId = (SELECT ProductCode FROM Inserted)
        INSERT INTO Audit (NewValue, OldValue, WindowsUser, ProductCode,
UpdateDate) VALUES (@newValue, @oldValue, @UserName, @itemId, @UpdateDate)
    END
END
```

## View/Delete the trigger

The trigger can be viewed by expanding the products table in SQL. From here you can delete the trigger if needed.



## View Audit Table

```
select * from Audit
```

	NewValue	Oldvalue	ABMUser	ProductCode	UpdateDate
1	Free of GST	10% GST	ABM\Laura.Wing	14500	2011-09-26 11:58:03.370
2	Imports	10% GST	ABM\Laura.Wing	12920	2011-09-26 12:08:02.590

\*THIS BULLETIN IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. The information contained in this document represents the current view of Advanced Business Manager on the issues discussed as of the date of publication. Because Advanced Business Manager must respond to change in market conditions, it should not be interpreted to be a commitment on the part of Advanced Business Manager and Advanced Business Manager cannot guarantee the accuracy of any information presented after the date of publication.

INFORMATION PROVIDED IN THIS DOCUMENT IS PROVIDED 'AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND FREEDOM FROM INFRINGEMENT.

The user assumes the entire risk as to the accuracy and the use of this document.