

```

USE [TESTDimensions]
GO
/***** Object:  StoredProcedure [OCC].[Price Book]    Script Date: 21/08/2020 20:48:10
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- =====
-- Author:          Stephen Moore
-- Create date: 10/03/2020
-- Description:      PriceBook data for OCC

-- Version:         1.1
--CASE statement added for Type 1
--12/03/2020, S.Moore

-- Version          2
-- inclusion of accounts piped data
-- 19/05/2020, S.Moore, T.Randall

-- Version          2.1
-- inclusion of extra columns to match pricelist TSD update;from notes to
quoteinfo.contractor and corrections to site and type
--22/05/2020, T.Randall

-- Version          3
-- inclusion of code to replicate new desired format for Leviton's OIC format
-- Accommodates 'USD' currencies
-- 3.1 Change to pricelistId to varchar (so that blanks are supported)
-- 3.2 Updates to replace '/' with '_' and ' ' with '' in intems.itemNumber
-- 3.3 Addition of soldToAddress columns

-- =====
ALTER PROCEDURE [OCC].[PriceBook]

AS
BEGIN

    SET NOCOUNT ON;

    ----- V1
    -- Write to table

    DECLARE @num int
    DECLARE @StrNum varchar(2)
    DECLARE @sql nvarchar(MAX)

    IF EXISTS (SELECT * FROM DataAssist.INFORMATION_SCHEMA.TABLES
WHERE TABLE_NAME = N'StockPrice' AND TABLE_CATALOG='DataAssist' and TABLE_SCHEMA='dbo')
    BEGIN
        DROP TABLE DataAssist.dbo.StockPrice
    
```

END

```
CREATE TABLE DataAssist.dbo.StockPrice (
    [priceListId] VARCHAR(5) --V3.1
    , [name] VARCHAR(100)
    , [accounts] nvarchar(MAX)
    , [type] VARCHAR(50) DEFAULT ''
    , [startDate] VARCHAR(50) DEFAULT ''
    , [endDate] VARCHAR(50) DEFAULT ''
    , [currency] VARCHAR(5)
    , [sourcesystem] VARCHAR(50)
    , [batchNumber] VARCHAR(50) DEFAULT ''
    , [siteID] VARCHAR(50) DEFAULT 'jcc'
    , [shippingAddress.address1] VARCHAR(50) DEFAULT ''
    , [shippingAddress.address2] VARCHAR(50) DEFAULT ''
    , [shippingAddress.city] VARCHAR(50) DEFAULT ''
    , [shippingAddress.companyName] VARCHAR(50) DEFAULT ''
    , [shippingAddress.country] VARCHAR(50) DEFAULT ''
    , [shippingAddress.postalCode] VARCHAR(50) DEFAULT ''
    , [shippingAddress.state] VARCHAR(50) DEFAULT ''
    , [shippingAddress.county] VARCHAR(50) DEFAULT ''
    , [soldToAddress.address1] VARCHAR(50) DEFAULT ''
    , [soldToAddress.address2] VARCHAR(50) DEFAULT ''
    , [soldToAddress.city] VARCHAR(50) DEFAULT ''
    , [soldToAddress.companyName] VARCHAR(50) DEFAULT ''
    , [soldToAddress.country] VARCHAR(50) DEFAULT ''
    , [soldToAddress.postalCode] VARCHAR(50) DEFAULT ''
    , [soldToAddress.state] VARCHAR(50) DEFAULT ''
    , [soldToAddress.county] VARCHAR(50) DEFAULT ''
    , [notes] VARCHAR(5) DEFAULT ''
    , [quoteInfo.salesPerson] VARCHAR(50) DEFAULT ''
    , [quoteInfo.email] VARCHAR(50) DEFAULT '' --V2.1
    , [quoteInfo.project] VARCHAR(20) DEFAULT '' --V2.1
    , [quoteInfo.contractor] VARCHAR(20) DEFAULT '' --V2.1
    , [items.itemNumber] VARCHAR(50)
    , [items.price] DECIMAL(10,2)
    , [items.qty] INT DEFAULT 1,
    [keynum] FLOAT
)

SET @num=0

WHILE (@num<10)
BEGIN
    SET @num=@num+1
    SET @StrNum = CAST(@num as varchar)
    SET @sql =
        'INSERT INTO DataAssist.dbo.StockPrice ([priceListId], [name], [items.price],
        [currency],[sourcesystem], [items.itemNumber], [type], [keynum])
        SELECT
            ' + @StrNum + ' As priceListId,
            1.STK_USRPRCKEY'+ @StrNum + ' As PriceBook,
            CAST(STK_SELLPRICE' + @StrNum + ' AS numeric(10,2)) AS [Sell Price],
            CASE WHEN STK_S_CURRCDE' + @StrNum + ' = ''£'' THEN ''GBP'' ELSE STK_S_CURRCDE' + @StrNum
            + ' END AS [Currency Code], --Version 1.1
            ''EN_423'' AS [SourceSystem],
            ''EN_423_''+TRIM(STKCODE2) AS [itemNumber],
            ''account'',
```

```

' + @StrNum + ' As keynum
FROM
STK_STOCK_2
JOIN
LANG_USERFIELDS 1
ON 1.STK_USRPRCKEY'+ @StrNum + '=1.STK_USRPRCKEY'+ @StrNum
+ ' WHERE CAST(STK_SELLPRICE' + @StrNum + ' AS numeric(10,2)) <>0'

EXEC(@sql)

END

----- V2
-- Piped Accounts

DECLARE @accounts varchar(MAX)
DECLARE @pricekeys
TABLE
(
    pricekey int NOT NULL,
    accountname varchar(MAX) NOT NULL
);

SET @num=0

WHILE @num<10
BEGIN
    SET @num=@num+1
    SET @accounts=''
    SELECT @accounts=@accounts + CUCODE + '|' from SL_ACCOUNTS where
CU_PRICE_KEY = @num
    INSERT INTO @pricekeys(pricekey,accountname) SELECT @num,
SUBSTRING(@accounts, 0, LEN(@accounts) -1)
END

UPDATE DataAssist.dbo.StockPrice SET accounts = p.accountname
FROM @pricekeys p JOIN DataAssist.dbo.StockPrice s ON p.pricekey=s.pricelistid

----- V3
-- New format

--Provide USD with unique keynum
UPDATE DataAssist.dbo.StockPrice SET keynum = 100
WHERE currency='USD'

-- Create the MV table used for grouped, top records
DECLARE @StockHeader
TABLE
(
    [priceListid] VARCHAR(5) --v3.1
    ,[name] VARCHAR(100)
    ,[accounts] nvarchar(MAX)
    ,[type] VARCHAR(50) DEFAULT ''
    ,[currency] VARCHAR(5)
    ,[sourcesystem] VARCHAR(50)
    ,[siteID] VARCHAR(50) DEFAULT 'jcc'
    ,[items.itemNumber] VARCHAR(50)

```

```

        ,[items.price] DECIMAL(10,2)
        ,[items.qty] INT DEFAULT 1,
        [keynum] FLOAT
    );

    -- Insert the grouped records into the MVT
    INSERT INTO @StockHeader (priceListid,name,accounts, [type], currency,
sourcesystem, [siteID], [items.itemNumber],[items.price], [items.qty], [keynum] )

        SELECT
            priceListid,name,accounts,[type], currency, sourcesystem, siteid,

            MAX([items.itemNumber])[items.itemNumber],MAX([items.price])[items.price],MAX([ite
ms.qty])[items.qty],
            CASE
                WHEN currency='GBP' THEN CAST(PriceListid as int)*0.91
                WHEN currency='USD' THEN 100*0.91
            END As keyNum
        from DataAssist.dbo.StockPrice
        GROUP BY priceListid,name,accounts,[type],currency, sourcesystem, siteid

-- Update the keynum field in StockPrice table from the records found in the MVT
UPDATE DataAssist.dbo.StockPrice
SET [keynum]=h.[keyNum]
FROM DataAssist.dbo.StockPrice s
JOIN @StockHeader h
ON s.[name]= h.[name]
AND s.[currency]=h.[currency]
AND s.[items.itemNumber]=h.[items.itemNumber]
AND s.[items.qty]=h.[items.qty]
AND s.[items.qty]=h.[items.qty]

-- Update StockPrice specific columns to blanks where the keynums are the pricelistID's
(not the new keynums from MVT)
UPDATE DataAssist.dbo.StockPrice
SET [pricelistid]='', [name]='', [accounts]='', [type]='', [currency]='',
[sourcesystem]='', [siteID]=''
WHERE [keyNum] IN (1,2,3,4,5,6,7,8,9,10,100)

-----V3.2

--Update [items.itemNumber] to remove blanks and replace slashes with underscores
UPDATE DataAssist.dbo.StockPrice
SET [items.itemNumber] = REPLACE(REPLACE([items.itemNumber], '/', '-'), ' ', '_')

END

```