ALTER procedure [dbo].[pmHATBLoadDexTempTables] @I\_cVendorTemp char(40) = NULL,

 @I\_cDocumentTemp char(40) = NULL,

 @I\_cApplyTemp char(40) = NULL,

 @I\_cVendorListTemp char(40) = NULL,

 @I\_cStartVendorID char(15) = NULL,

 @I\_cEndVendorID char(15) = NULL,

 @I\_cStartVendorName char(65) = NULL,

 @I\_cEndVendorName char(65) = NULL,

 @I\_cStartClassID char(15) = NULL,

 @I\_cEndClassID char(15) = NULL,

 @I\_cStartUserDefined char(15) = NULL,

 @I\_cEndUserDefined char(15) = NULL,

 @I\_cStartPaymentPriority char(3) = NULL,

 @I\_cEndPaymentPriority char(3) = NULL,

 @I\_tExcludeNoActivity tinyint = NULL,

 @I\_tExcludeZeroBalanceVendors tinyint = NULL,

 @I\_tExcludeFullyPaidTrxs tinyint = NULL,

 @I\_tExcludeCreditBalance tinyint = NULL,

 @I\_tExcludeMultiCurrency tinyint = NULL,

 @O\_iErrorState int = NULL output as declare @TRUE smallint,

 @FALSE smallint,

 @vVendorTable varchar(255),

 @cExcludeFullyPaidTrxs char(1),

 @cExcludeMulticurrency char(1),

 @cStartVendorID char(32),

 @cEndVendorID char(32),

 @cStartVendorName char(132),

 @cEndVendorName char(132),

 @cStartClassID char(32),

 @cEndClassID char(32),

 @cStartUserDefined char(32),

 @cEndUserDefined char(32),

 @cStartPaymentPriority char(8),

 @cEndPaymentPriority char(8),

 @tLoopControl tinyint,

 @iStatus int,

 @iError int

select

 @iStatus = 0,

 @O\_iErrorState = 0

select

 @TRUE = 1,

 @FALSE = 0 while @tLoopControl is NULL begin

select

 @tLoopControl = 1 if @I\_cVendorTemp is NULL

 or @I\_cDocumentTemp is NULL

 or @I\_cApplyTemp is NULL

 or @I\_cVendorListTemp is NULL

 or @I\_cStartVendorID is NULL

 or @I\_cEndVendorID is NULL

 or @I\_cStartVendorName is NULL

 or @I\_cEndVendorName is NULL

 or @I\_cStartClassID is NULL

 or @I\_cEndClassID is NULL

 or @I\_cStartUserDefined is NULL

 or @I\_cEndUserDefined is NULL

 or @I\_cStartPaymentPriority is NULL

 or @I\_cEndPaymentPriority is NULL

 or @I\_tExcludeNoActivity is NULL

 or @I\_tExcludeZeroBalanceVendors is NULL

 or @I\_tExcludeFullyPaidTrxs is NULL

 or @I\_tExcludeCreditBalance is NULL

 or @I\_tExcludeMultiCurrency is NULL begin

select

 @O\_iErrorState = 20962 break end if @I\_cVendorListTemp = ''

select

 @vVendorTable = ' PM00200 A with (NOLOCK) ' else

select

 @vVendorTable = @I\_cVendorListTemp + ' T join PM00200 A with (NOLOCK) on T.VENDORID = A.VENDORID ' if (@I\_tExcludeNoActivity = @FALSE)

 and (

 @I\_tExcludeZeroBalanceVendors = @FALSE

 ) begin Exec @iStatus = smFormatStringsForExecs @I\_cStartVendorID,

 @cStartVendorID output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cEndVendorID,

 @cEndVendorID output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cStartVendorName,

 @cStartVendorName output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cEndVendorName,

 @cEndVendorName output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cStartClassID,

 @cStartClassID output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cEndClassID,

 @cEndClassID output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cStartUserDefined,

 @cStartUserDefined output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cEndUserDefined,

 @cEndUserDefined output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cStartPaymentPriority,

 @cStartPaymentPriority output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break Exec @iStatus = smFormatStringsForExecs @I\_cEndPaymentPriority,

 @cEndPaymentPriority output,

 @O\_iErrorState output

select

 @iError = @@error if @iStatus = 0

 and @iError <> 0

select

 @iStatus = @iError if (@iStatus <> 0)

 or (@O\_iErrorState <> 0) break exec(

 'insert into ' + @I\_cVendorTemp + ' select A.VENDORID, A.VENDNAME, A.VNDCLSID, A.USERDEF1, A.PYMNTPRI, A.VENDORID from ' + @vVendorTable + ' where A.VENDORID >= ' + @cStartVendorID + ' and A.VENDORID <= ' + @cEndVendorID + ' and A.VENDNAME >= ' + @cStartVendorName + ' and A.VENDNAME <= ' + @cEndVendorName + ' and A.VNDCLSID >= ' + @cStartClassID + ' and A.VNDCLSID <= ' + @cEndClassID + ' and A.USERDEF1 >= ' + @cStartUserDefined + ' and A.USERDEF1 <= ' + @cEndUserDefined + ' and A.PYMNTPRI >= ' + @cStartPaymentPriority + ' and A.PYMNTPRI <= ' + @cEndPaymentPriority

 ) end else begin exec(

 'insert into ' + @I\_cVendorTemp + ' select distinct A.VENDORID, A.VENDNAME, A.VNDCLSID, A.USERDEF1, A.PYMNTPRI, A.VENDORID from ' + @vVendorTable + ', #DocumentTEMP where A.VENDORID = #DocumentTEMP.VendorID'

 ) end if (

 @I\_tExcludeZeroBalanceVendors = @TRUE

 )

 and (@I\_tExcludeCreditBalance = @TRUE) begin exec(

 'delete ' + @I\_cVendorTemp + ' where (select sum(CurrentTrxAmount) from #DocumentTEMP where ' + @I\_cVendorTemp + '.VENDORID = #DocumentTEMP.VendorID ) <= 0.00'

 ) end else if @I\_tExcludeZeroBalanceVendors = @TRUE begin exec(

 'delete ' + @I\_cVendorTemp + ' where (select sum(CurrentTrxAmount) from #DocumentTEMP where ' + @I\_cVendorTemp + '.VENDORID = #DocumentTEMP.VendorID ) = 0.00'

 ) end else if @I\_tExcludeCreditBalance = @TRUE begin exec(

 'delete ' + @I\_cVendorTemp + ' where (select sum(CurrentTrxAmount) from #DocumentTEMP where ' + @I\_cVendorTemp + '.VENDORID = #DocumentTEMP.VendorID ) < 0.00'

 ) end

update

 #DocumentTEMP set AgingPeriod = 0 where DocumentType = 7 select @cExcludeFullyPaidTrxs = convert(char(1),@I\_tExcludeFullyPaidTrxs) select @cExcludeMulticurrency = convert(char(1),@I\_tExcludeMultiCurrency) exec('insert into ' + @I\_cDocumentTemp + ' select VendorID, VoucherNumber, case WHEN (DocumentType < 6) THEN 0 ELSE 1 end, DocumentNumber, DocumentType, case WHEN (DocumentType < 4 or DocumentType = 7) THEN DocumentAmount ELSE -DocumentAmount end, case WHEN (DocumentType < 4) THEN DiscountTakenAmount ELSE 0.00 end, DocumentDate, DueDate, DiscountDate, TrxSource, CurrentTrxAmount, case WHEN (AgingPeriod = 1) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, case WHEN (AgingPeriod = 2) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, case WHEN (AgingPeriod = 3) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, case WHEN (AgingPeriod = 4) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, case WHEN (AgingPeriod = 5) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, case WHEN (AgingPeriod = 6) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, case WHEN (AgingPeriod = 7) THEN case WHEN (DocumentType < 4) THEN (DocumentAmount - RevalDocAmountAdj - DiscountTakenAmount - WriteoffAmount) ELSE (CurrentTrxAmount) end ELSE 0.00 end, (DiscountAmountAvail + RevalDiscAmountAvailAdj), AgingPeriod, case WHEN (DocumentType < 4) THEN WriteoffAmount ELSE 0.00 end, case WHEN (DocumentType < 4) THEN (rtrim(VoucherNumber) + ''0'') ELSE (rtrim(VoucherNumber) + ''1'') end, '''', GLPostingDate, case WHEN (' + @cExcludeMulticurrency + ' = 1) THEN '''' ELSE CurrencyID end, CurrencyIndex, ExchangeRate, case WHEN (DocumentType < 4) THEN OrigDocumentAmount ELSE (-1 \* OrigDocumentAmount) end, case WHEN (DocumentType < 4) THEN OrigDiscTakenAmount ELSE 0.00 end, OrigCurrentTrxAmount, case WHEN (AgingPeriod = 1) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, case WHEN (AgingPeriod = 2) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, case WHEN (AgingPeriod = 3) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, case WHEN (AgingPeriod = 4) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, case WHEN (AgingPeriod = 5) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, case WHEN (AgingPeriod = 6) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, case WHEN (AgingPeriod = 7) THEN case WHEN (DocumentType < 4) THEN (OrigDocumentAmount - OrigDiscTakenAmount - OrigWriteoffAmount) ELSE OrigCurrentTrxAmount end ELSE 0.00 end, OrigDiscAmountAvail, case WHEN (DocumentType < 4) THEN OrigWriteoffAmount ELSE 0.00 end, DenominationExchangeRate, MCTransactionState from #DocumentTEMP where ' + @cExcludeFullyPaidTrxs + ' = 0 or ( ' + @cExcludeFullyPaidTrxs + ' = 1 and CurrentTrxAmount <> 0.00)') if (@I\_tExcludeZeroBalanceVendors = @TRUE or @I\_tExcludeCreditBalance = @TRUE) and (@I\_tExcludeMultiCurrency = @FALSE) exec('delete ' + @I\_cDocumentTemp + ' where NOT EXISTS( select 1 from ' + @I\_cVendorTemp + ' A where A.VENDORID = ' + @I\_cDocumentTemp + '.VENDORID)') exec('insert into ' + @I\_cApplyTemp + ' select ApplyToVoucherNumber, ApplyToDocumentType, VoucherNumber, DocumentType, case WHEN (AgingPeriod = 1) THEN -AppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 2) THEN -AppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 3) THEN -AppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 4) THEN -AppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 5) THEN -AppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 6) THEN -AppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 7) THEN -AppliedAmount ELSE 0.00 end, AppliedAmount, Posted, CurrencyID, CurrencyIndex, ExchangeRate, OrigAppliedAmount, case WHEN (AgingPeriod = 1) THEN -OrigAppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 2) THEN -OrigAppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 3) THEN -OrigAppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 4) THEN -OrigAppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 5) THEN -OrigAppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 6) THEN -OrigAppliedAmount ELSE 0.00 end, case WHEN (AgingPeriod = 7) THEN -OrigAppliedAmount ELSE 0.00 end, (-1 \* RealizedGainLossAmount), DenominationExchangeRate, MCTransactionState from #ApplyTEMP') end return(@iStatus)

 GO