ALTER TABLE tbl\_PFRepriceOEMs

ADD Comments [nvarchar](max) NULL;

Stored Procedures:

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[CheckForProcessorList] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[CheckForProcessorList]

@sComments nvarchar(250) null

as

BEGIN

if (@sComments='GetListDate')

BEGIN

select top 1 LastUpdatedOn from tbl\_pfcompletelist order by 1 desc

END

if (@sComments='TruncateList')

BEGIN

Truncate table tbl\_pfcompletelist

END

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[CountOfDeletedComments] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[CountOfDeletedComments]

(

@ExecutionID int =NULL,

@WeekID nvarchar(100) = NULL,

@OEMName nvarchar(500) = NULL,

@Comments nvarchar(500) = NULL

)

As

Begin

if(@Comments = 'OHRNationalAcademic')

Begin

select count(\*) from tbl\_PfRawOEM with(noLock)

where ExecutionId = @ExecutionID

AND WeekID = @WeekID

AND [OEM Name] = @OEMName

AND DeletedComments = 'National Academic'

AND DeletedYN = 1

AND [Processor Model Validation Result] <> 'Fail'

AND [Total Physical RAM Validation Result] <> 'Fail'

AND [Primary Disk Total Capacity Validation Result] <> 'Fail'

AND [High End Licensable (SKU) Validation] <> 'Fail'

--AND [ScreenSizeDiagonal Validation Result] <> 'Fail'

--AND [ScreenResolutionDPI Validation Result] <> 'Fail'

AND [Display Size Validation Result] <> 'Fail'

AND [Display Resolution validation Result] <> 'Fail'

AND [Primary Disk Type Validation Result] <> 'Fail'

AND [Optical Disk Drive Type Validation Result] <> 'Fail'

AND [Digitizer Support Validation Result] <> 'Fail'

AND ([OHR Form Factor Sub Class Validation Result] = 'Fail'

OR [OHR Form Factor Validation Result] = 'Fail'

OR [OHR Touch Validation Result] = 'Fail'

OR [OHR Screen Size Validation Result] = 'Fail')

END

ELSE

BEGIN

select count(\*) from tbl\_PfRawOEM with(noLock)

where ExecutionId = @ExecutionID

AND WeekID = @WeekID

AND [OEM Name] = @OEMName

AND DeletedComments = @Comments

AND DeletedYN = 1

END

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[DeleteGMContactDetails] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <07-09-2017>

-- Description: <This method is used to Delete GM Contact details into database>

-- =============================================

Create ProcEDURE [dbo].[DeleteGMContactDetails]

(

@GMID int = Null ,

@Status varchar(100) = null output

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

update tbl\_GMContactList

set DeletedYN = 1,

LastUpdatedOn = getdate()

where ID=@GMID

set @Status = 'OEM Deleted Successfully'

select @Status as Status1

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[EmptyCancelledFile] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[EmptyCancelledFile]

AS

BEGIN

SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

where ID =1

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetCommentsFromtbl\_PFRepriceOEMs] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create ProcEDURE [dbo].[GetCommentsFromtbl\_PFRepriceOEMs]

-- Add the parameters for the stored procedure here

@WeekID NVARCHAR(MAX),

@OEMName NVARCHAR(MAX),

@ProcessExecutionID int

AS

BEGIN

SET NOCOUNT ON;

select Comments

FROM tbl\_PFRepriceOEMs

WHERE WeekID =@WeekID AND ReportID =@OEMName AND ProcessExecutionID = @ProcessExecutionID

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetDeleteCommentsForRawFile] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- EXEC [GetDeleteCommentsForRawFile] 1353,20170730,'ASUSTEK COMPUTER INC.'

ALTER ProcEDURE [dbo].[GetDeleteCommentsForRawFile]

@ExecutionID INT = NULL,

@WeekID INT = NULL,

@OEMName NVARCHAR(200) = NULL

AS

BEGIN

SET NOCOUNT ON;

--Select \* from tbl\_PFRawOEM

SELECT COUNT(\*) as [Count],DeletedComments FROM tbl\_PFRawOEM

WHERE ExecutionID = @ExecutionID AND

WeekID = @WeekID AND

[OEM Name] = @OEMName AND

DeletedYN = 1 --AND

--DeletedComments = 'Deleted based on HardwarePPT Rules'

GROUP BY DeletedComments

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetFailuresFromRawFile] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[GetFailuresFromRawFile]

(

@OEMName varchar(255) = NULL,

@ExecutionID int = NULL,

@WeekID int = NULL,

@DeletedComments varchar(255) = NULL

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

select 'Primary Disk Total Capacity Fail,'+ [Primary Disk Total Capacity] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [Primary Disk Total Capacity Validation Result] = 'Fail'

Group by [Primary Disk Total Capacity]

--UNION

--select 'Processor Model Fail,'+ [Processor Model] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

--where WEEKID = @WeekID

--and ExecutionID = @ExecutionID

--and [OEM Name] = @OEMName

--and DeletedYN = 0

--and [Processor Model Validation Result] = 'Fail'

--Group by [Processor Model]

UNION

select 'Total Physical RAM Fail,'+ [Total Physical RAM] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [Total Physical RAM Validation Result] = 'Fail'

Group by [Total Physical RAM]

UNION

select 'Primary Display Size Fail,'+ [Primary Display Size (Inch)] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [Display Size Validation Result] = 'Fail'

Group by [Primary Display Size (Inch)]

UNION

select 'Primary Disk Type Fail,'+ [Primary Disk Type] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [Primary Disk Type Validation Result] = 'Fail'

Group by [Primary Disk Type]

UNION

select 'Optical Disk Drive Type Fail,'+ [Optical Disk Drive Type] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [Optical Disk Drive Type Validation Result] = 'Fail'

Group by [Optical Disk Drive Type]

UNION

select 'Digitizer Support Fail,'+ [Digitizer Support] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [Digitizer Support Validation Result] = 'Fail'

Group by [Digitizer Support]

UNION

select 'High End Licensable (SKU) Validation Fail,'+ [High End Licensable (SKU) Validation] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 0

and [High End Licensable (SKU) Validation] = 'Fail'

Group by [High End Licensable (SKU) Validation]

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetNAORecordsFromRawFileForOEM] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Kishore Kumar Pechetti>

-- Create date: <08-07-2017>

-- Description: < This method is used to get NAO deleted records for OEM>

-- exec GetNAORecordsFromRawFileForOEM 'ACER INDIA (PVT) LTD.',1523,20170827,'National Academic'

-- =============================================

ALTER ProcEDURE [dbo].[GetNAORecordsFromRawFileForOEM]

(

@OEMName varchar(255) = NULL,

@ExecutionID int = NULL,

@WeekID int = NULL,

@DeletedComments varchar(255) = NULL

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

if(@DeletedComments = 'National Academic')

BEGIN

( SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

WHERE [OEM Name] = @OEMName

AND WeekID=@WeekID

AND ExecutionID=@ExecutionID

AND DeletedComments = @DeletedComments

)

Except(

SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

WHERE [OEM Name] = @OEMName

AND WeekID=@WeekID

AND ExecutionID=@ExecutionID

AND [Processor Model Validation Result] <> 'Fail'

AND [Total Physical RAM Validation Result] <> 'Fail'

AND [Primary Disk Total Capacity Validation Result] <> 'Fail'

--AND [ScreenSizeDiagonal Validation Result] <> 'Fail'

--AND [ScreenResolutionDPI Validation Result] <> 'Fail'

AND [Display Size Validation Result] <> 'Fail'

AND [Display Resolution validation Result] <> 'Fail'

AND [Primary Disk Type Validation Result] <> 'Fail'

AND [Optical Disk Drive Type Validation Result] <> 'Fail'

AND [Digitizer Support Validation Result] <> 'Fail'

AND ([OHR Form Factor Sub Class Validation Result] = 'Fail'

OR [OHR Form Factor Validation Result] = 'Fail'

OR [OHR Touch Validation Result] = 'Fail'

OR [OHR Screen Size Validation Result] = 'Fail')

AND DeletedComments = @DeletedComments

)

ORDER BY PKID

END

ELSE if(@DeletedComments = 'OHR Only Fail')

BEGIN

SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

WHERE [OEM Name] = @OEMName

AND WeekID=@WeekID

AND ExecutionID=@ExecutionID

--AND DeletedComments = @DeletedComments

AND DeletedYN = 0

AND [Processor Model Validation Result] <> 'Fail'

AND [Total Physical RAM Validation Result] <> 'Fail'

AND [Primary Disk Total Capacity Validation Result] <> 'Fail'

AND [High End Licensable (SKU) Validation] <> 'Fail'

--AND [ScreenSizeDiagonal Validation Result] <> 'Fail'

--AND [ScreenResolutionDPI Validation Result] <> 'Fail'

AND [Display Size Validation Result] <> 'Fail'

AND [Display Resolution validation Result] <> 'Fail'

AND [Primary Disk Type Validation Result] <> 'Fail'

AND [Optical Disk Drive Type Validation Result] <> 'Fail'

AND [Digitizer Support Validation Result] <> 'Fail'

AND ([OHR Form Factor Sub Class Validation Result] = 'Fail'

OR [OHR Form Factor Validation Result] = 'Fail'

OR [OHR Touch Validation Result] = 'Fail'

OR [OHR Screen Size Validation Result] = 'Fail')

ORDER BY PKID

END

ELSE

BEGIN

SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

WHERE [OEM Name] = @OEMName

AND WeekID=@WeekID

AND ExecutionID=@ExecutionID

AND DeletedComments = @DeletedComments

ORDER BY PKID

END

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetOEMWeeklySettings] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <04-09-2017>

-- Description: <This method is used to GetOEMWeeklySetting details frrom database>

--

-- =============================================

Create Proc [dbo].[GetOEMWeeklySettings]

AS

BEGIN

SELECT

ID,TopNOEM,

[Weekday],

[Repricing OEM Eligibility],

[CreatedBy],

[ModifiedBy]

FROM dbo.[tbl\_OEMWeeklySettings]

where Active=0

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetProcessorListName] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[GetProcessorListName]

@DeviceCategory nvarchar(250) null

AS

BEGIN

SELECT [DevelopedMarket(DM)] from tbl\_PFProcesorList where DeviceCategory =@DeviceCategory

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetRawFileDataWithOutOHROnlyFail] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

--sp\_helptext GetRawFileDataWithOutOHROnlyFail

-- =============================================

-- Author: <Kishore Kumar Pechetti>

-- Create date: <08-07-2017>

-- Description: < This method is used to get OHR Only Fail for OEM>

-- exec GetRawFileDataWithOutOHROnlyFail 'INFORMFINANS SERVICE',1449,20170806,'National Academic'

-- =============================================

ALTER ProcEDURE [dbo].[GetRawFileDataWithOutOHROnlyFail]

(

@OEMName varchar(255) = NULL,

@ExecutionID int = NULL,

@WeekID int = NULL

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

(SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

WHERE [OEM Name] = @OEMName

AND WeekID=@WeekID

AND ExecutionID=@ExecutionID

AND DeletedYN = 0

)

Except(

SELECT

[Source]

,[OEM Name] as 'Partner (Sold To)'

,[OEM Soldto #] as 'Partner Number (Sold To)'

,[Agreement#] as 'Agreement #'

,[PKID]

,[Licensable (SKU)] as 'Licensable Name'

,[Licensable Part #] as 'Licensable Part Number'

,[Device Category]

,[Collection OS]

,[Product Family] as 'Product Family Name'

,[Market]

,[Royalty]

,[Fulfilment Date] as 'Fulfillment Date'

,[CBR Submission Date] as 'Build Report Received Date'

,[Processed Date]

,[Last Process Date] as 'Last Processed Date'

,[Processor Model]

,[Processor Model Validation Result]

,[Total Physical RAM]

,[Total Physical RAM Validation Result]

,[Primary Disk Total Capacity]

,[Primary Disk Total Capacity Validation Result]

,[Internal Primary Display Size Physical Vertical (MM)] as 'Manufacturing Screen Size Physical Y (MM)'

,[Internal Primary Display Size Physical Horizontal (MM)] as 'Manufacturing Screen Size Physical H (MM)'

,[Primary Display Size (Inch)] as 'Manufacturing Screen Size (Inch)'

,[Display Size Validation Result] as 'ScreenSizeDiagonal Validation Result'

,[Internal Primary Display Resolution Vertical] as 'Display Resolution Vertical'

,[Internal Primary Display Resolution Horizontal] as 'Display Resolution Horizontal'

,[Display Resolution Validation Result] as 'ScreenResolutionDPI Validation Result'

,[Primary Disk Type]

,[Primary Disk Type Validation Result]

,[Optical Disk Drive Type]

,[Optical Disk Drive Type Validation Result]

,[Digitizer Support] as 'Digitizer Support Name'

,[Digitizer Support Validation Result]

,[High End Licensable (SKU) Validation]

,[OHR Form Factor]

,[OHR Form Factor Validation Result]

,[OHR Form Factor Sub Class]

,[OHR Form Factor Sub Class Validation Result]

,[Chassis Type Form Factor]

,[Chassis Type Form Factor Sub Class]

,[Chassis Type Validation Result]

,[OHR Touch]

,[OHR Touch Validation Result]

,[OHR Screen Size]

,[OHR Screen Size Validation Result]

,[OHR Model Number] as 'OHR PC Model Number'

FROM tbl\_PFRawOEM WITH(NOLOCK)

WHERE [OEM Name] = @OEMName

AND WeekID=@WeekID

AND ExecutionID=@ExecutionID

AND [Processor Model Validation Result] <> 'Fail'

AND [Total Physical RAM Validation Result] <> 'Fail'

AND [Primary Disk Total Capacity Validation Result] <> 'Fail'

AND [High End Licensable (SKU) Validation] <> 'Fail'

--AND [ScreenSizeDiagonal Validation Result] <> 'Fail'

--AND [ScreenResolutionDPI Validation Result] <> 'Fail'

AND [Display Size Validation Result] <> 'Fail'

AND [Display Resolution validation Result] <> 'Fail'

AND [Primary Disk Type Validation Result] <> 'Fail'

AND [Optical Disk Drive Type Validation Result] <> 'Fail'

AND [Digitizer Support Validation Result] <> 'Fail'

AND ([OHR Form Factor Sub Class Validation Result] = 'Fail'

OR [OHR Form Factor Validation Result] = 'Fail'

OR [OHR Touch Validation Result] = 'Fail'

OR [OHR Screen Size Validation Result] = 'Fail')

)

ORDER BY PKID

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetRawFileDetailsForNotifications] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[GetRawFileDetailsForNotifications]

@ExecutionID int =NULL,

@OEMName NVARCHAR(200) =NULL,

@WeekID int =NULL

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Insert statements for procedure here

SELECT top 1 [OEM Soldto #]+','+[Agreement#] from tbl\_PFRawOEM WITH(NOLOCK) WHERE ExecutionID =@ExecutionID AND [OEM Name] = @OEMName AND WeekID=@WeekID

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[GetReports] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <05-09-2017>

-- Description: <This method is used to GetReport details frrom database>

--

-- =============================================

Create Proc [dbo].[GetReports]

AS

BEGIN

SELECT

WeekID,OEMName,OEMType,PKIDCount,[Status]

FROM dbo.tbl\_PFLogOEM

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[InsertGMContactDetails] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <07-09-2017>

-- Description: <This method is used to insert GMContact details into database>

-- =============================================

Create ProcEDURE [dbo].[InsertGMContactDetails]

(

@GMID int,

@TVOName varchar(100) ,

@GMContactName varchar(100) ,

@TOAlias varchar(100) ,

@CCAlias varchar(100) ,

@Status varchar(100) = null output

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Insert statements for procedure here

BEGIN

INSERT INTO dbo.tbl\_GMContactList

([ID],[TVOName],[GMContactName],TOAlias,CCAlias,DeletedYN)

VALUES

(@GMID,@TVOName,@GMContactName,@TOAlias,@CCAlias,0)

set @Status = 'Record Created Successfully'

select @Status as Status1

END

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[InsertOEMData] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[InsertOEMData]

(

@OEMsName nvarchar(100),

@MLANumber nvarchar(100),

@SoldTo nvarchar(100),

@SoldToLocation nvarchar(100),

@Vat decimal = Null,

@ToAlias nvarchar(100),

@CCAlias nvarchar(100),

@CreatedBy nvarchar(100),

@Status nvarchar(100) = null output

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

BEGIN

IF EXISTS(select 1 from tbl\_MassCollabOEMContacts where [OEMs on Mass Collab]=@OEMsName and [MLA Number]=@MLANumber and SoldTo=@SoldTo and SoldToLocation=@SoldToLocation and [VAT%]=@Vat and DeletedYN=0)

BEGIN

set @Status = 'Record already exists'

select @Status as Status1

END

ELSE

BEGIN

INSERT INTO dbo.[tbl\_MassCollabOEMContacts]

([OEMs on Mass Collab],[MLA Number],SoldTo,SoldToLocation,[VAT%],Toalias,CCalias,DeletedYN,CreatedOn,CreatedBy)

VALUES

(@OEMsName,@MLANumber,@SoldTo,@SoldToLocation,@Vat,@ToAlias,@CCAlias,0,GETDATE(),@CreatedBy)

set @Status = 'OEM created successfully'

select @Status as Status1

END

END

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[ProcessorListValidationLogic] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[ProcessorListValidationLogic]

@WeekID NVARCHAR(250) NULL,

@ExecutionID INT NULL,

@OEMName NVARCHAR(250) NULL

AS

BEGIN

DECLARE @Table1 TABLE (ID INT)

Insert into @Table1(ID )

select DISTINCT PFRAW.ID from tbl\_pfRawoem PFRAW

join tbl\_PFProcesorList PL on PFRAW.[Device Category] = PL.[RawFileDeviceCategory]

join tbl\_PFCompleteList CL on CL.ProcessorList = 'List '+PL.[DevelopedMarket(DM)]

where PFRAW.WEEKID = @WeekID

and PFRAW.ExecutionID = @ExecutionID

and PFRAW.[OEM Name] = @OEMName

and PFRAW.DeletedYN = 0

and PFRAW.[Processor Model Validation Result] = 'Fail'

and CHARINDEX(CL.[Model#],PFRAW.[Processor Model]) <> 0

UPDATE tbl\_pfRawoem

SET DeletedYN = 1 , DeletedComments = 'Processor Model Fail'

WHERE ID IN(SELECT ID from @Table1)

Begin

select 'Processor Model Fail,'+ [Processor Model] as ColumnValue, count(\*) as TotalCount from tbl\_pfRawoem

where WEEKID = @WeekID

and ExecutionID = @ExecutionID

and [OEM Name] = @OEMName

and DeletedYN = 1

and DeletedComments = 'Processor Model Fail'

and [Processor Model Validation Result] = 'Fail'

Group by [Processor Model]

END

END

--exec [ProcessorListValidationLogic] '20170903','35', 'DELL INC.'

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[UpdateGMContactDetails] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <07-09-2017>

-- Description: <This method is used to update GM Contact details into database>

-- =============================================

Create ProcEDURE [dbo].[UpdateGMContactDetails]

(

@GMID int,

@TVOName varchar(100) ,

@GMContactName varchar(100) ,

@TOAlias varchar(100) ,

@CCAlias varchar(100) ,

@LastUpdatedBy varchar(100) = NULL,

@Status varchar(100) = null output

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Insert statements for procedure here

update dbo.tbl\_GMContactList

set

[TVOName] = @TVOName,

[GMContactName] = @GMContactName,

TOAlias = @TOAlias,

CCAlias = @CCAlias,

LastUpdatedOn = Getdate(),

LastUpdatedBy = @LastUpdatedBy

where ID=@GMID

set @Status = 'Record Updated Successfully'

select @Status as Status1

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[UpdateOEM] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[UpdateOEM]

(

@OEMID int = Null,

@OEMsName nvarchar(100),

@MLANumber nvarchar(100),

@SoldTo nvarchar(100),

@SoldToLocation nvarchar(100),

@Vat decimal = Null,

@ToAlias nvarchar(100),

@CCAlias nvarchar(100),

@LastUpdatedBy nvarchar(100),

@Status nvarchar(100) = null output

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

update [tbl\_MassCollabOEMContacts]

set [OEMs on Mass Collab] = @OEMsName,

[MLA Number] = @MLANumber,

SoldTo = @SoldTo,

SoldToLocation = @SoldToLocation,

[VAT%] = @Vat,

Toalias = @ToAlias,

CCalias = @CCAlias,

DeletedYN = 0,

LastUpdatedOn = Getdate(),

LastUpdatedBy = @LastUpdatedBy

where ID= @OEMID

set @Status = 'OEM Updated Successfully'

select @Status as Status1

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[UpdateOEMSettings] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <04-09-2017>

-- Description: <This method is used to update OEM Settngs details into database>

-- =============================================

Create ProcEDURE [dbo].[UpdateOEMSettings]

(

@ID int,

@TopNOEM int ,

@Weekday varchar(100),

@RepricingEligibility int,

@LastUpdatedBy varchar(100) = NULL,

@Status varchar(100) = null output

)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Insert statements for procedure here

update dbo.[tbl\_OEMWeeklySettings]

set [TopNOEM] = @TopNOEM ,

[Weekday] = @Weekday,

[Repricing OEM Eligibility]=@RepricingEligibility,

ModifiedBy = @LastUpdatedBy

where ID=@ID

set @Status = 'Record Updated Successfully'

select @Status as Status1

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[UpdateStatusBasedOnTableName] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[UpdateStatusBasedOnTableName]

-- Add the parameters for the stored procedure here

@TableName nvarchar(max),

@WeekID nvarchar(max),

@ReportID nvarchar(max),

@Status nvarchar(max),

@ProcessExecutionID int,

@Comments nvarchar(max)

as

BEGIN

IF(@TableName='tbl\_PFReconciliation')

BEGIN

UPDATE tbl\_PFReconciliation SET Status =@Status where ExecutionID = @ProcessExecutionID and WeekID=@WeekID

END

ELSE IF(@TableName= 'tbl\_PFRepriceOEMs')

BEGIN

UPDATE tbl\_PFRepriceOEMs SET Status =@Status, Comments =@Comments where ProcessExecutionID = @ProcessExecutionID and WeekID=@WeekID and ReportID =@ReportID

END

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_DisplayGMContactsList] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: <Gopal Krishna k>

-- Create date: <07-09-2017>

-- Description: <This method is used to Display GMContactList details frrom database>

-- =============================================

Create ProcEDURE [dbo].[usp\_DisplayGMContactsList]

AS

BEGIN

Select ID,

TVOName,

GMContactName ,

TOAlias,

ISNULL(CCAlias,'') CCAlias

from tbl\_GMContactList where DeletedYN=0

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_DisplayOEMMasterDetails] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[usp\_DisplayOEMMasterDetails]

AS

BEGIN

Select ID,

[OEMs on Mass Collab],

ISNULL([MLA Number],'') [MLA Number],

ISNULL(SoldTo,'') SoldTo,

ISNULL(SoldToLocation,'') SoldToLocation,

ISNULL([VAT%],0) [VAT%] ,

Toalias,

ISNULL(CCalias,'') CCalias

from tbl\_MassCollabOEMContacts where DeletedYN=0

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_GetSoldToAAndMLA] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[usp\_GetSoldToAAndMLA]

(

@OEMName nvarchar(500) = NULL

)

AS

BEGIN

SET NOCOUNT ON;

SELECT [SoldTo] + ',' +[MLA Number] from tbl\_MassCollabOEMContacts where [SoldTo] = @OEMName

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[usp\_ReturnWeekIDs] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[usp\_ReturnWeekIDs]

AS

BEGIN

Select Distinct WeekID From tbl\_PFLogOEM where DeletedYN=0

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[ValidateOHRFail] Script Date: 9/11/2017 12:37:44 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER ProcEDURE [dbo].[ValidateOHRFail]

@ProcessExecutionID INT = NULL,

@WeekID NVARCHAR(500) = NULL,

@ReportID NVARCHAR(500) = NULL

AS

BEGIN

SET NOCOUNT ON;

--Select \* from tbl\_PFRawOEM

UPDATE tbl\_PFScrubbedData SET DeletedYN=1, [RequiredLicensable(SKU)] = 'Cancelled',

[Required Licensable Part No.] = 'NA', [Required Device Category] = 'NA',

LastUpdatedOn=GETDATE()

WHERE ReportID = @ReportID AND ProcessExecutionID=@ProcessExecutionID AND WeekID=@WeekID

AND [Processor Model Validation Result] <> 'Fail'

AND [Total Physical RAM Validation Result] <> 'Fail'

AND [Primary Disk Total Capacity Validation Result] <> 'Fail'

AND [Display Size Validation Result] <> 'Fail'

AND [Display Resolution validation result] <> 'Fail'

AND [Primary Disk Type Validation Result] <> 'Fail'

AND [Optical Disk Drive Type Validation Result] <> 'Fail'

AND [Digitizer Support Validation Result] <> 'Fail'

AND [Chassis Type Validation Result] <> 'Fail'

AND ([OHR Form Factor Sub Class Validation Result] = 'Fail' OR [OHR Form Factor Validation Result] = 'Fail' OR [OHR Screen Size Validation Result] = 'Fail')

END

GO

/\*\*\*\*\*\* Object: Table [dbo].[tbl\_PFProcesorList] Script Date: 9/11/2017 3:30:08 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[tbl\_PFProcesorList](

[ID] [int] IDENTITY(1,1) NOT NULL,

[DeviceCategory] [nvarchar](250) NOT NULL,

[DevelopedMarket(DM)] [nvarchar](100) NOT NULL,

[EmergingMarket(DM)] [nvarchar](100) NOT NULL,

[ODM] [nvarchar](100) NOT NULL,

[DeletedYN] [bit] NULL,

[LastUpdatedOn] [datetime] NULL,

[LastUpdatedBy] [nvarchar](100) NULL,

[RawFileDeviceCategory] [nvarchar](250) NULL

)

GO

SET IDENTITY\_INSERT [dbo].[tbl\_PFProcesorList] ON

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (1, N'WW Home Small Tablet', N'A', N'A', N'A', 0, CAST(0x0000A7E600784870 AS DateTime), N'V-VENNSU', N'Windows 10 Home Small Tablet | 2 in 1')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (2, N'WW Home for Stick PC', N'B', N'B', N'B', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'Windows 10 Home PC on a Stick')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (3, N'WW Home Entry Notebook', N'C', N'C', N'C', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'Windows 10 Home Entry Notebook')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (4, N'Home Value Notebook', N'C', N'C', N'n/a', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'Windows 10 Home Value Notebook')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (5, N'Home Value Desktop / AiO', N'D', N'D', N'n/a', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'Windows 10 Home Value Desktops | AiOs')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (6, N'WW Home Value 2 in 1', N'E', N'E', N'n/a', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'Windows 10 Home Value Tablet | 2 in 1')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (7, N'WW Home Entry Desktop/AiO', N'F', N'F', N'F', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'Windows 10 Home Entry Desktops | AiOs')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (8, N'WW Home High End', N'G', N'G', N'n/a', 0, CAST(0x0000A7E60079DC8E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (9, N'STF Entry Desktop/AiO', N'H', N'H', N'n/a', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (10, N'STF Value 2 in 1', N'I', N'I', N'n/a', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (11, N'STF Value Notebook', N'J', N'J', N'n/a', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (12, N'STF Pro Tablet and Small Tablet', N'K', N'K', N'n/a', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (13, N'STF High End', N'L', N'L', N'n/a', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (14, N'WW Pro Small Tablet', N'M', N'M', N'M', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'Windows 10 Pro Small Tablet | 2 in 1')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (15, N'WW Pro Tablet & 2 in 1', N'M', N'M', N'M', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'Windows 10 Pro Tablet | 2 in 1')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (16, N'WW Pro Entry Notebook', N'N', N'N', N'N', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (17, N'WW Pro Entry Desktop / AiO', N'O', N'O', N'O', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'Windows 10 Pro Entry Desktops | AiOs')

INSERT [dbo].[tbl\_PFProcesorList] ([ID], [DeviceCategory], [DevelopedMarket(DM)], [EmergingMarket(DM)], [ODM], [DeletedYN], [LastUpdatedOn], [LastUpdatedBy], [RawFileDeviceCategory]) VALUES (18, N'WW Pro Entry Stick PC', N'P', N'P', N'P', 0, CAST(0x0000A7E6007B721E AS DateTime), N'V-VENNSU', N'')

SET IDENTITY\_INSERT [dbo].[tbl\_PFProcesorList] OFF

ALTER TABLE [dbo].[tbl\_PFProcesorList] ADD DEFAULT ((0)) FOR [DeletedYN]

GO

ALTER TABLE [dbo].[tbl\_PFProcesorList] ADD DEFAULT (getdate()) FOR [LastUpdatedOn]

GO