```sql
USE [TOC_Jwellery]
GO
/****** Object:  StoredProcedure [dbo].[USP_Buffer_Increase_Decrease]    Script Date: 4/10/2022 10:03:23 AM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER  PROCEDURE [dbo].[USP_Buffer_Increase_Decrease]

@RequestType varchar(200) =NULL,
@StoreCode varchar(200) =NULL,
@Ean varchar(200) =NULL,
@XmlInput AS XML = NULL
--exec [USP_Buffer_Increase_Decrease] 'Buffer_Increase_Decrease','CWH','MAHARAJA_KW2079/2900',''
AS
BEGIN
    SET NOCOUNT ON;
    BEGIN TRY
    DECLARE @Recommended_Buffer as Int=0;
    DECLARE @Noofdays_minus as Int=-7;
    DECLARE @Replenishment_time as Int;
    DECLARE @Counter_date as date;
    DECLARE @Increase_Buffer_Value as Int=1;
    DECLARE @Decrease_Buffer_Value as Int=1;
    SET @StoreCode=@StoreCode;--'4'; ---this will change when job creates @parameter
    SET @Ean=@Ean;--'8903953000011';  ---this will change when job creates  @parameter

    IF(@RequestType='Buffer_Increase_Decrease')
    BEGIN

      update Tbl_ArsMaster set Recommend_Buffer=0,Recommend='',Recommend_Date='',Warning='',
          [Status]= '' where Store_Code=@StoreCode AND Ean=@Ean;

    ---check Current Buffer value using store code and Ean code
     SELECT top 1 * INTO #tmp_ArsMaster from Tbl_ArsMaster where Ean=@Ean and Store_Code=@StoreCode;
    SET @Replenishment_time=(select top 1 Replenishment_time from Tbl_LocationMaster with(nolock) where Store_Code=@StoreCode);
    SET @Counter_date=(select cast(Recommend_Date as date) from #tmp_ArsMaster)
    SET @Counter_date=dateadd(dd,@Replenishment_time,@Counter_date);
     --looping only those record where  buffer value is 1 or 2

     --select Buffer from #tmp_ArsMaster where buffer>=1 and  buffer <=2
     IF((select count(*) from #tmp_ArsMaster where buffer>=1 and  buffer <=2)>0)
     BEGIN
     print'ajeet'
    DECLARE @SUM_Sale_Qty as Int=0;
    --find the total sale which has happened in the last 42 days
    SET @SUM_Sale_Qty=isnull((select SUM(sl.QTY) as Sale_Qty from Tbl_Transcation_master sl   where  sl.Store_Code=@StoreCode AND sl.Ean=@Ean
      and sl.[Date] >=DATEADD(dd, @Noofdays_minus, (select max([Date]) from Tbl_Transcation_master))),0);

      --check sold 2 or more pices
      IF(@SUM_Sale_Qty >=2)
      BEGIN
       print'2'
      ---get sale data min and max dates
       select  min(sl.[Date]) as mindate, max(sl.[Date]) as maxdate INTO #tempDate from Tbl_Transcation_master sl
       where  sl.[Date] >=  DATEADD(dd, @Noofdays_minus, (select max([Date]) from Tbl_Transcation_master  )) --where  Store_Code=@StoreCode AND Ean=@Ean

       DECLARE @Inv_NoofDays as int=0;
       ---check no of days SOH available in Inventory >0
       SET @Inv_NoofDays=( select count(*) from Tbl_Inventory_Details where SOH >0
         and Store_Code=@StoreCode AND Ean=@Ean  and Inv_date > (select mindate from #tempDate) and Inv_date < (select maxdate from #tempDate));


        --get dbm policy
         SELECT top 1 * INTO #tmp_DbmPolicy FROM Tbl_Dbm_policy where Dbm_Policy_Name=(select DBM_Policy from #tmp_ArsMaster);

         ---check no of days inventory was >0
         IF(@Inv_NoofDays <= (select Consumption_Frequency from #tmp_DbmPolicy) and (select Store_site_buffer_color from #tmp_ArsMaster) in('Red','Black'))
         BEGIN
         IF(@Inv_NoofDays >0)
         BEGIN
         SET @Recommended_Buffer=(select isnull(Buffer,0) as Buffer_old from #tmp_ArsMaster)+ @Increase_Buffer_Value;

         ---update Recommend buffer in Ars Master table for respective storeocde and ean code
         update Tbl_ArsMaster set Recommend_Buffer=ROUND(@Recommended_Buffer,0),Recommend='Increase',Recommend_Date=GETDATE(),
        [Status]= case when (select isnull(Buffer,0) as Buffer_old from #tmp_ArsMaster) !=@Recommended_Buffer then 'Pending' ELSE
                  (select [Status] from #tmp_ArsMaster) END
         where Store_Code=@StoreCode AND Ean=@Ean;
         END
         END
         --ELSE
```

```
        --BEGIN
        -- update Tbl_ArsMaster set Recommend_Buffer=0,Recommend='',Recommend_Date='',
        -- [Status]= '' where Store_Code=@StoreCode AND Ean=@Ean;
        --END

       -- select @Recommended_Buffer as Recommended_Buffer;

      DROP TABLE #tmp_DbmPolicy;

      DROP TABLE #tempDate;
     END
     ELSE IF(@SUM_Sale_Qty <=1 and (select [Buffer] from #tmp_ArsMaster)=2)--check sold 0 or 1 pices for skus when buffer is=2
     BEGIN
      print'3'
      DECLARE @Inv_NoofDays_available as int=0;
       ---get data from inventory tables
       SET @Inv_NoofDays_available=( select count(*) from Tbl_Inventory_Details where SOH >0
         and Store_Code=@StoreCode AND Ean=@Ean  and Inv_date > DATEADD(dd, @Noofdays_minus, (select max(Inv_date) from Tbl_Inventory_Details )));


       --get dbm policy
        SELECT top 1 * INTO #tmp_DbmPolicy_ FROM Tbl_Dbm_policy where Dbm_Policy_Name=(select DBM_Policy from #tmp_ArsMaster)

        ---check no of days inv.
        IF(@Inv_NoofDays_available > (select Small_Buffer_Reduction from #tmp_DbmPolicy_)  and (select Store_site_buffer_color from #tmp_ArsMaster) ='Green')
        BEGIN
        SET @Recommended_Buffer=(select isnull(Buffer,0) as Buffer_old from #tmp_ArsMaster)- @Decrease_Buffer_Value;

        ---update Recommend buffer in Ars Master table for respective storeocde and ean code
         update Tbl_ArsMaster set Recommend_Buffer=ROUND(@Recommended_Buffer,0),Recommend='Decrease' ,Recommend_Date=GETDATE(),
       [Status]= case when (select isnull(Buffer,0) as Buffer_old from #tmp_ArsMaster) !=@Recommended_Buffer then 'Pending' ELSE
                 (select [Status] from #tmp_ArsMaster) END where Store_Code=@StoreCode AND Ean=@Ean;
        END
        --ELSE
        --BEGIN
        -- update Tbl_ArsMaster set Recommend_Buffer=0,Recommend='',Recommend_Date='',
        -- [Status]= '' where Store_Code=@StoreCode AND Ean=@Ean;
        --END

      DROP TABLE #tmp_DbmPolicy_;

     END
    END --end ofrequest type condition
     ELSE IF((select [Buffer] from #tmp_ArsMaster) > 2)
     BEGIN

    DECLARE @Warning as varchar(100)='';
      SELECT top 1 * INTO #tmp_Dbm_large_buffer FROM Tbl_Dbm_policy where Dbm_Policy_Name=(select DBM_Policy from #tmp_ArsMaster);

       select top 1 * INTO #temp_ArsMaster_history from Tbl_ArsMaster_history with(nolock) where  1=1 --cast(Created_at as date)=cast(getdate()-1 as date)
       and Ean=@Ean and Store_Code=@StoreCode order by Created_at desc;



     ------------- RED ZONE BUFFER LOGIC----------------------------------------
     DECLARE @Top_Red_zone as decimal(18,2);
      DECLARE @Tot_accum_needed as decimal(18,2);
      DECLARE @YesterdyTop_Red_zone as decimal(18,2);
      DECLARE @Yesterday_Penetration as decimal(18,2);
      DECLARE @Cum_Penetration as decimal(18,2);
       DECLARE @today_Penetration as decimal(18,2);


      SET @Top_Red_zone=(cast((select [Buffer] from #tmp_ArsMaster) as float)*(select Buffer_Red_BP_Level from #tmp_Dbm_large_buffer))/100;
      SET @Tot_accum_needed=cast((@Top_Red_zone*(select Increase_Trigger from #tmp_Dbm_large_buffer))/100 as float);

      SET @YesterdyTop_Red_zone=(cast((select [Buffer] from #temp_ArsMaster_history) as float)*(select Buffer_Red_BP_Level from #tmp_Dbm_large_buffer));

      SET @Yesterday_Penetration=(@YesterdyTop_Red_zone-isnull((select isnull(SOH,0)  from Tbl_Inventory_Details  where SOH >0
         and Store_Code=@StoreCode AND Ean=@Ean  and Inv_date =cast(getdate()-2 as date)),0));

         IF(@Yesterday_Penetration < 0)
         BEGIN
         SET @Yesterday_Penetration=0;
         END

         SET @today_Penetration=(@Top_Red_zone-isnull((select isnull(SOH,0)  from Tbl_Inventory_Details   where SOH >0
         and Store_Code=@StoreCode AND Ean=@Ean   and Inv_date =cast(getdate()-1 as date)),0));

         IF(@today_Penetration <0)
         BEGIN
         SET @today_Penetration=0;
```

```
        END

         SET @Cum_Penetration=@today_Penetration+@Yesterday_Penetration;

         DECLARE @SUM_of_CONS as decimal(18,2)= (select isnull(sum(cast(QTY as decimal(18,2))),0)  from Tbl_Transcation_master  where QTY >0
        and Store_Code=@StoreCode AND Ean=@Ean   and [Date] =cast(getdate()-(ROUND(@Replenishment_time/2,0)) as date));

    DECLARE @virtual_pipe_Cnt as decimal(18,2)= (select count(*) from Tbl_Inventory_Details     where  isnull(SOH,0)+isnull(cast(GIT as decimal(18,2)),0) >=
        (cast((select [Buffer] from #tmp_ArsMaster) as float)*(select Buffer_Green_BP_Level from #tmp_Dbm_large_buffer))
        and SOH >0      and Store_Code=@StoreCode AND Ean=@Ean  and Inv_date > cast(getdate()-@Replenishment_time as date));

       IF(@SUM_of_CONS >(select [Buffer] from #tmp_ArsMaster))
       BEGIN
            SET  @Warning='Sharp Demand Increase';
       END
       IF(@virtual_pipe_Cnt =0)
       BEGIN
        SET  @Warning='Improper Replenishment';

       END


        ----check cum pentration with curr buffer------------
        IF((@Cum_Penetration >=@Tot_accum_needed) and (select Store_site_buffer_color from #tmp_ArsMaster) in('Red','Black'))
        BEGIN

        DECLARE @Rec_Buffer as decimal(18,2);
        SET @Rec_Buffer=cast((select [Buffer] from #tmp_ArsMaster) as float)*(select Increase_Factor from #tmp_Dbm_large_buffer)/100;
        print @StoreCode;
         print @Ean;

         ---update Recommend buffer in Ars Master table for respective storeocde and ean code
         update Tbl_ArsMaster set Recommend_Buffer=ROUND(isnull(Buffer,2)+@Rec_Buffer,0),
         Recommend='Increase' ,Recommend_Date=GETDATE(),Cooling_period='No',Warning=@Warning,
       [Status]= case when (select isnull(Buffer,0) as Buffer_old from #tmp_ArsMaster) !=@Rec_Buffer then 'Pending'
       ELSE (select [Status] from #tmp_ArsMaster) END
                 where Store_Code=@StoreCode AND Ean=@Ean;
                -- print 'Aq'
        END
--      ELSE
       --BEGIN
```