

Note and Coding (or Script) - 1

Note

1

```
USE [Syteline9_App]
GO

/***** Object: StoredProcedure [dbo].[TNG_UpdatePriceToContractPriceSp]    Script Date:
8/24/2022 10:03:15 AM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO
```

```
/*****
Created by: Ken Wu
Date: 12/28/2020
*****/
```

```
CREATE PROCEDURE [dbo].[TNG_UpdatePriceToContractPriceSp]
AS
declare @SetSite SiteType
select @SetSite = site from parms_mst with (readuncommitted)
exec dbo.SetSiteSp @SetSite, null

-- BEGIN TRAN

update [dbo].[coitem]
set coitem.price = itemcustprice.cont_price, coitem.price_conv =
itemcustprice.cont_price
from [dbo].[coitem]
inner join [itemcustprice] on [coitem].[co_cust_num] = [itemcustprice].[cust_num] and
[coitem].[item] = [itemcustprice].[item]
where UPPER(Rtrim(LTrim([coitem].stat))) = 'O' -- and
UPPER(RTRIM(LTRIM([coitem].[co_cust_num]))) = 'C000185'

-- COMMIT TRAN
GO
```

```
USE [Syteline9_App]
GO

/***** Object: StoredProcedure [dbo].[TNG_UpdatePriceToZeroSp]    Script Date:
8/24/2022 10:04:16 AM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO
```

```
/*****
Created by: Ken Wu
Date: 12/29/2020
*****/
```

```
CREATE PROCEDURE [dbo].[TNG_UpdatePriceToZeroSp]
```

```
AS
declare @SetSite SiteType
select @SetSite = site from parms_mst with (readuncommitted)
exec dbo.SetSiteSp @SetSite, null

update [dbo].[coitem]
set price = 0, price_conv = 0
where UPPER(Rtrim(LTrim(stat))) = 'O' -- and co_cust_num = 'c000077'
GO
```

-- Please confirm that you receive this latest version.

-- Ken Good below

-- Enhanced by Ken Wu 8/24/2022
DECLARE @Invdate DateType

SET @Invdate = GETDATE()

```
SELECT item,
       sum(qty_on_hand) AS qoh
INTO #railloc
FROM itemloc_mst
-- WHERE item = 'INDhpNTpt' Comment by Ken 8/23/2022
-- AND loc = 'rail' Comment by Ken 8/23/2022
GROUP BY item
```

```
SELECT i.item
       ,i.ufprofile
       ,i.description
       ,iw.whse
       ,CASE
           WHEN i.item = 'INDhpNTpt'
               THEN sum(iw.qty_on_hand + isnull(r.qoh,0))
           ELSE sum(iw.qty_on_hand)
           END AS qty_on_hand
       ,CASE
           WHEN i.item = 'INDhpNTpt'
               THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.unit_cost)
           ELSE sum(iw.qty_on_hand * i.unit_cost)
           END AS value
       ,CASE
           WHEN i.item = 'INDhpNTpt'
               THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.matl_cost)
```

2

```

ELSE sum(iw.qty_on_hand * i.matl_cost)
END AS valuemat
,CASE
  WHEN i.item = 'INDhpNTpt'
    THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.lbr_cost)
  ELSE sum(iw.qty_on_hand * i.lbr_cost)
  END AS valuelab
,CASE
  WHEN i.item = 'INDhpNTpt'
    THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.vovhd_cost)
  ELSE sum(iw.qty_on_hand * i.vovhd_cost)
  END AS valuevov

```

```

INTO #current
FROM item_mst i
INNER JOIN itemwhse_mst iw ON i.item = iw.item
LEFT JOIN #railloc r ON r.item = i.item
GROUP BY i.item
       ,i.ufprofile
       ,i.description
       ,iw.whse

```

```

SELECT m.item
       ,m.whse
       ,sum(qty) AS qty_on_hand
       ,value = CASE
         WHEN (it.cost_method = 'A')
           THEN sum(qty * cost)
         ELSE sum(qty * it.unit_cost)
         END
       ,valuemat = CASE
         WHEN (it.cost_method = 'A')
           THEN sum(qty * m.matl_cost)
         ELSE sum(qty * it.matl_cost)
         END
       ,valuelab = CASE
         WHEN (it.cost_method = 'A')
           THEN sum(qty * m.lbr_cost)
         ELSE sum(qty * it.lbr_cost)
         END
       ,valuevov = CASE
         WHEN (it.cost_method = 'A')
           THEN sum(qty * m.vovhd_cost)
         ELSE sum(qty * it.vovhd_cost)
         END

```

```

INTO #tran
FROM matltran_mst m
INNER JOIN item_mst it ON m.item = it.item
WHERE trans_date >= @invdate
      AND m.trans_type <> 'c'
      AND m.loc <> '20'
GROUP BY m.item
       ,it.cost_method
       ,m.whse

```

```

-- Below query Added by Ken Wu 8/24/2022
select item,
whse,

```

```

sum(qty_rsvd) as qty_rsvd
into #whse_qty_rsvd
from itemloc_mst
-- where item = '1.0x5.50x144xWWxR063xNA' and whse = 'BLN1'
group by item, whse

```

```

SELECT c.item
       ,c.whse
       ,c.description
       ,c.ufprofile
       ,rsvd.qty_rsvd
       ,c.qty_on_hand - isnull(t.qty_on_hand,0) as Qty
       ,(c.qty_on_hand * i.unit_weight) - isnull((t.qty_on_hand * i.unit_weight),0) as
Lbs
       ,c.value - isnull(t.value,0) as Value
       ,(c.value - isnull(t.value,0))/Nullif(((c.qty_on_hand - isnull(t.qty_on_hand,0)) *
i.unit_weight),0) '$/Lb'
       ,c.valuemat - isnull(t.valuemat,0) as matvalue
       ,c.valuelab - isnull(t.valuelab,0) as labvalue
       ,c.valuevov - isnull(t.valuevov,0) as vovvalue

```

```

FROM #current c
LEFT JOIN #tran t ON c.item = t.item and c.whse = t.whse
INNER JOIN item_mst i ON c.item = i.item
INNER JOIN #whse_qty_rsvd rsvd ON rsvd.item = c.item and rsvd.whse = c.whse -- Added by
Ken Wu 8/24/2022
WHERE (
        c.qty_on_hand <> isnull(t.qty_on_hand, 0.0)
        OR c.value <> isnull(t.value, 0.0)
      )
      AND c.item NOT LIKE 'Pfrt%'
      -- AND c.item = '1.0x5.50x144xWWxR063xNA'
order by c.item, c.whse

```

DROP TABLE #current

DROP TABLE #tran

DROP TABLE #railloc

Drop Table #whse_qty_rsvd

-- Please confirm that receive the Email.

-- Thank You for the testing!

-- Ken

Use Syteline9_App -- Added by Ken Wu 8/25/2022

-- Enhanced by Ken Wu 8/25/2022 - Final Version 1

3

```
select #tempTotal_Syteline.co_num_syteline,
#tempTotal_Syteline.Total_Qty_x_Price_Syteline,
#tempTotal_Qry.Total_Qty_x_Price_Qry,
#tempTotal_Qry.co_num_qry,
#tempTotal_Syteline.Total_Qty_x_Price_Syteline - #tempTotal_Qry.Total_Qty_x_Price_Qry as
Price_Different
from #tempTotal_Syteline
full outer join #tempTotal_Qry
on RTRIM(LTRIM(#tempTotal_Syteline.co_num_syteline)) =
RTRIM(LTRIM(#tempTotal_Qry.co_num_qry))
where #tempTotal_Syteline.Total_Qty_x_Price_Syteline <>
#tempTotal_Qry.Total_Qty_x_Price_Qry
order by #tempTotal_Syteline.co_num_syteline ASC
```

Good_4_13_2022

Use Syteline9_App
Go

```
-- Created by Ken Wu 4/12/2022
select co.cust_num,
custaddr.[name],
co.co_num,
co_ship.co_line,
coitem.item,
item.unit_weight,
co_ship.qty_invoiced,
co_ship.qty_shipped,
Format(co_ship.ship_date, 'yyyy-MM-dd') ship_date,
co_ship.price Ship_Price,
co_ship.price Invoice_Price,
co_ship.cost,
co_ship.matl_cost,
co_ship.lbr_cost,
co_ship.vovhd_cost,
co_ship.fovhd_cost,
coitem.whse,
item.ufprofile,
item.ufcolor,
item.Uflength
into #temp_main
from co_ship_mst co_ship
inner join coitem_mst coitem on RTRIM(LTRIM(co_ship.co_num)) =
RTRIM(LTRIM(coitem.co_num)) and
RTRIM(LTRIM(co_ship.co_line)) = RTRIM(LTRIM(coitem.co_line))
inner join item_mst item on RTRIM(LTRIM(coitem.item)) = RTRIM(LTRIM(item.item))
inner join co_mst co on RTRIM(LTRIM(co.co_num)) = RTRIM(LTRIM(coitem.co_num))
inner join custaddr_mst custaddr on RTRIM(LTRIM(co.cust_num)) =
RTRIM(LTRIM(custaddr.cust_num))
and
custaddr.cust_seq = '0'
where Format(co_ship.ship_date, 'yyyy-MM-dd') >= '2022-01-01' -- 'yyyy-MM-dd'
and
Format(co_ship.ship_date, 'yyyy-MM-dd') <= '2022-01-31' -- 'yyyy-MM-dd'
Order by co.cust_num, custaddr.name, co.co_num, co_ship.co_line, coitem.item, ship_date
```

```
select inv_hdr_mst.co_num,
inv_item_mst.co_line,
inv_hdr_mst.cust_num,
inv_item_mst.price Invoice_Price,
inv_hdr_mst.inv_date,
inv_item_mst.[qty_invoiced]
into #temp_detail
from [dbo].inv_hdr_mst
inner join inv_item_mst on inv_item_mst.co_num = [inv_hdr_mst].co_num
inner join custaddr_mst on custaddr_mst.cust_num = inv_hdr_mst.cust_num
inner join [co_ship_mst] on [co_ship_mst].co_num = [inv_hdr_mst].co_num
inner join item_mst on item_mst.item = inv_item_mst.item
where custaddr_mst.cust_seq = 0
and RTRIM(LTRIM(inv_hdr_mst.site_ref)) = RTRIM(LTRIM(inv_item_mst.site_ref))
and RTRIM(LTRIM(inv_hdr_mst.inv_num)) = RTRIM(LTRIM(inv_item_mst.inv_num))
and RTRIM(LTRIM(inv_hdr_mst.inv_seq)) = RTRIM(LTRIM(inv_item_mst.inv_seq))
and RTRIM(LTRIM(inv_hdr_mst.cust_num)) =
RTRIM(LTRIM(custaddr_mst.cust_num))
and
(Format(inv_hdr_mst.inv_date, 'yyyy-MM-dd') >= '2022-01-01'
and
Format(inv_hdr_mst.inv_date, 'yyyy-MM-dd') <= '2022-01-31')
and
-- inv_item_mst.price >= 0
--
and
RTRIM(LTRIM(co_ship_mst.co_line)) = RTRIM(LTRIM(inv_item_mst.co_line)) =
-- Added 4/11/2022

update #temp_main
set #temp_main.Invoice_Price = #temp_detail.Invoice_Price
from #temp_main
inner join dbo.#temp_detail
on (RTRIM(LTRIM(#temp_main.cust_num)) = RTRIM(LTRIM(#temp_detail.cust_num))) and
(RTRIM(LTRIM(#temp_main.co_num)) = RTRIM(LTRIM(#temp_detail.co_num))) and
(RTRIM(LTRIM(#temp_main.co_line)) = RTRIM(LTRIM(#temp_detail.co_line)))

select *
into #temp_main2
from #temp_main
Order by cust_num, [name], co_num, co_line, item, ship_date

select *
from #temp_main2
order by co_num ASC

select co_num as co_num_qry, sum(Invoice_Price * qty_invoiced) Total_Qty_x_Price_Qry
into #tempTotal_Qry
from #temp_main2
group by co_num
order by co_num Desc

select inv_item_mst.[co_num] as co_num_syteline,
SUM(inv_item_mst.[qty_invoiced] * inv_item_mst.[price]) Total_Qty_x_Price_Syteline
into #tempTotal_Syteline
from [dbo].[inv_item_mst]
inner join [dbo].inv_hdr_mst on inv_item_mst.co_num = [inv_hdr_mst].co_num
where (Format(inv_hdr_mst.inv_date, 'yyyy-MM-dd') >= '2022-01-01'
and
```



```
Format(inv_hdr_mst.inv_date, 'yyyy-MM-dd') <= '2022-01-31')
group by inv_item_mst.[co_num]
order by inv_item_mst.[co_num] DESC

select #tempTotal_Syteline.co_num_syteline,
#tempTotal_Syteline.Total_Qty_x_Price_Syteline, #tempTotal_Qry.Total_Qty_x_Price_Qry,
#tempTotal_Qry.co_num_qry
from #tempTotal_Syteline
full outer join #tempTotal_Qry
on RTRIM(LTRIM(#tempTotal_Syteline.co_num_syteline)) =
RTRIM(LTRIM(#tempTotal_Qry.co_num_qry))
order by #tempTotal_Syteline.co_num_syteline ASC

Drop Table #temp_main
Drop Table #temp_detail
Drop Table #temp_main2
Drop Table #tempTotal_Qry
Drop Table #tempTotal_Syteline
```

```
select invhdr.inv_num,
invhdr.site_ref,
invhdr.inv_date,
invhdr.cust_po,
invhdr.co_num,
invitm.co_line,
invitm.item,
invitm.qty_invoiced,
invitm.price price_item,
invhdr.price price_invoice
from inv_hdr_mst invhdr
inner join inv_item_mst invitm
on
invhdr.inv_num = invitm.inv_num AND invhdr.site_ref = invitm.site_ref AND invhdr.inv_seq
= invitm.inv_seq
where (Format(invhdr.inv_date, 'yyyy-MM-dd') >= '2022-01-01'
and
Format(invhdr.inv_date, 'yyyy-MM-dd') <= '2022-01-31')
order by invhdr.co_num ASC
```

Sure_Work_4_13_2022

Use Syteline9_App
Go

```
-- Created by Ken Wu 4/12/2022
select co.cust_num,
custaddr.[name],
co.co_num,
co_ship.co_line,
coitem.item,
item.unit_weight,
co_ship.qty_invoiced,
co_ship.qty_shipped,
```

```
Format(co_ship.ship_date, 'yyyy-MM-dd') ship_date,
co_ship.price Ship_Price,
co_ship.price Invoice_Price,
co_ship.cost,
co_ship.matl_cost,
co_ship.lbr_cost,
co_ship.vovhd_cost,
co_ship.fovhd_cost,
coitem.whse,
item.ufprofile,
item.ufcolor,
item.Uflength
into #temp_main
from co_ship_mst co_ship
inner join coitem_mst coitem on RTRIM(LTRIM(co_ship.co_num)) =
RTRIM(LTRIM(coitem.co_num)) and
RTRIM(LTRIM(co_ship.co_line)) = RTRIM(LTRIM(coitem.co_line))
inner join item_mst item on RTRIM(LTRIM(coitem.item)) = RTRIM(LTRIM(item.item))
inner join co_mst co on RTRIM(LTRIM(co.co_num)) = RTRIM(LTRIM(coitem.co_num))
inner join custaddr_mst custaddr on RTRIM(LTRIM(co.cust_num)) =
RTRIM(LTRIM(custaddr.cust_num))
and
custaddr.cust_seq = '0'
where Format(co_ship.ship_date, 'yyyy-MM-dd') >= '2022-01-01' -- 'yyyy-MM-dd'
and
Format(co_ship.ship_date, 'yyyy-MM-dd') <= '2022-01-31' -- 'yyyy-MM-dd'
Order by co.cust_num, custaddr.name, co.co_num, co_ship.co_line, coitem.item, ship_date

select invhdr.inv_num,
invhdr.site_ref,
invhdr.inv_date,
invhdr.cust_po,
invhdr.co_num,
invitm.co_line,
invitm.item,
invitm.qty_invoiced,
invitm.price price_item,
-- invhdr.price price_invoice,
invhdr.cust_num
into #temp_detail
from inv_hdr_mst invhdr
inner join inv_item_mst invitm
on
invhdr.inv_num = invitm.inv_num AND invhdr.site_ref = invitm.site_ref AND invhdr.inv_seq
= invitm.inv_seq
where (Format(invhdr.inv_date, 'yyyy-MM-dd') >= '2022-01-01'
and
Format(invhdr.inv_date, 'yyyy-MM-dd') <= '2022-01-31')
order by invhdr.co_num ASC

update #temp_main
set #temp_main.Invoice_Price = #temp_detail.price_item
from #temp_main
inner join dbo.#temp_detail
on (RTRIM(LTRIM(#temp_main.cust_num)) = RTRIM(LTRIM(#temp_detail.cust_num))) and
(RTRIM(LTRIM(#temp_main.co_num)) = RTRIM(LTRIM(#temp_detail.co_num))) and
(RTRIM(LTRIM(#temp_main.co_line)) = RTRIM(LTRIM(#temp_detail.co_line)))
```


5

```
select ite.item, itemlo.whse, itemlo.qty_rsvd
from item_mst ite
inner join itemloc_mst itemlo
on ite.item = itemlo.item
where itemlo.qty_rsvd > 0
group by itemlo.whse, ite.item, itemlo.qty_rsvd
```

```
BEGIN TRANSACTION
SET TRANSACTION ISOLATION LEVEL READ COMMITTED
```

```
select i.item,
       i.ufprofile,
       i.description,
       itemloc.qty_rsvd,
       itemloc.whse,
       sum(iw.qty_on_hand)+sum(iw.qty_trans) as qty_on_hand,
       sum(iw.qty_on_hand * i.unit_cost)+sum(iw.qty_trans * i.unit_cost) as value,
       sum(iw.qty_on_hand * i.matl_cost)+sum(iw.qty_trans * i.matl_cost) as
valuemat,
       sum(iw.qty_on_hand * i.lbr_cost)+sum(iw.qty_trans * i.lbr_cost) as valuelab,
       sum(iw.qty_on_hand * i.vovhd_cost)+sum(iw.qty_trans * i.vovhd_cost) as
valuevov
       /* sum(iw.qty_on_hand * i.cur_u_cost)+sum(iw.qty_trans * i.cur_u_cost) as
Cur_value,
       sum(iw.qty_on_hand * i.cur_matl_cost)+sum(iw.qty_trans * i.cur_matl_cost) as
Cur_valuemat,
       sum(iw.qty_on_hand * i.cur_lbr_cost)+sum(iw.qty_trans * i.cur_lbr_cost) as
Cur_valuelab,
       sum(iw.qty_on_hand * i.cur_vovhd_cost)+sum(iw.qty_trans * i.cur_vovhd_cost)
as Cur_valuevov */
into #current
from item_mst i
inner join itemwhse_mst iw on i.item = iw.item
inner join itemloc_mst itemloc on i.item = itemloc.item
where itemloc.qty_rsvd > 0
group by itemloc.whse, i.item, i.ufprofile, i.description, itemloc.qty_rsvd
```

```
select m.item,
       sum(qty) as qty_on_hand,
       value = case when (it.cost_method = 'A') then
sum(qty * cost)
       else sum(qty * it.unit_cost)
       end,
       valuemat = case when (it.cost_method = 'A') then
sum(qty * m.matl_cost)
       else sum(qty * it.matl_cost)
       end,
       valuelab = case when (it.cost_method = 'A') then
sum(qty * m.lbr_cost)
       else sum(qty * it.lbr_cost)
       end,
       valuevov = case when (it.cost_method = 'A') then
sum(qty * m.vovhd_cost)
       else sum(qty * it.vovhd_cost)
```

```
end
into #tran
from matltran_mst m
inner join item_mst it on m.item = it.item
group by m.item, it.cost_method
```

```
Select      c.item,
            c.[description] as [Description],
            c.ufprofile as Ufprofile,
            c.qty_on_hand as Qty,
            c.value as Value,
            c.valuemat as Matvalue,
            c.valuelab as LabValue,
            c.valuevov as VovValue,
            t.qty_on_hand as tranqtyonhand,
            t.value as tranValue,
            t.valuemat as tranValueMat,
            t.valuelab as tranValueLab,
            t.valuevov as tranValueVov, */
/*
--      i.unit_weight,
            c.qty_rsvd as Qty_rsvd,
            c.whse as Whse,
            c.qty_on_hand * unit_weight as Lbs,
            Case when c.qty_on_hand = 0 then 0 else c.value / c.qty_on_hand end as
DollarPerLbs
            --      (ISNULL(qty_on_hand,0) * ISNULL(unit_weight,0)) - (ISNULL(tranqtyonhand, 0)
* ISNULL(unit_weight,0)) as Lbs
            --      ((ISNULL(value,0) - ISNULL(tranValue,0)) / ((ISNULL(qty_on_hand,0) -
ISNULL(tranqtyonhand, 0)) * ISNULL(unit_weight,0))) END as DollarPerLbs,
-- into #InVasDateData
from #current c
left join #tran t on c.item = t.item
inner join item_mst i on c.item = i.item
```

```
drop table #current
drop table #tran
-- drop table #InVasDateData
-- drop table #FinalResult
```

```
COMMIT TRANSACTION
```

Good on 8-23-2022

```
-- y
DECLARE @Invdate DateTime
SET @Invdate = GETDATE() --ChangeDateHere
```

6

```

SELECT item,
-- qty_rsvd,
sum(qty_rsvd) as qty_rsvd,
sum(qty_on_hand) AS qoh
INTO #railloc
FROM itemloc_mst
-- WHERE item = 'INDhpNTpt' Comment by Ken 8/23/2022
-- AND loc = 'rail' Comment by Ken 8/23/2022
GROUP BY item --, qty_rsvd -- Added by Ken 8/23/2022

SELECT i.item
,i.ufprofile
,i.description
,iw.whse
,r.qty_rsvd -- Added by Ken 8/23/2022
,CASE
WHEN i.item = 'INDhpNTpt'
THEN sum(iw.qty_on_hand + isnull(r.qoh,0))
ELSE sum(iw.qty_on_hand)
END AS value
,CASE
WHEN i.item = 'INDhpNTpt'
THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.unit_cost)
ELSE sum(iw.qty_on_hand * i.unit_cost)
END AS value
,CASE
WHEN i.item = 'INDhpNTpt'
THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.matl_cost)
ELSE sum(iw.qty_on_hand * i.matl_cost)
END AS valuemat
,CASE
WHEN i.item = 'INDhpNTpt'
THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.lbr_cost)
ELSE sum(iw.qty_on_hand * i.lbr_cost)
END AS valuelab
,CASE
WHEN i.item = 'INDhpNTpt'
THEN sum(iw.qty_on_hand + isnull(r.qoh,0)) * max(i.vovhd_cost)
ELSE sum(iw.qty_on_hand * i.vovhd_cost)
END AS valuevov

INTO #current
FROM item_mst i
INNER JOIN itemwhse_mst iw ON i.item = iw.item
LEFT JOIN #railloc r ON r.item = i.item
GROUP BY i.item
,i.ufprofile
,i.description
,iw.whse
,r.qty_rsvd -- Added by Ken 8/23/2022

SELECT m.item
,m.whse
,sum(qty) AS qty_on_hand
,value = CASE
WHEN (it.cost_method = 'A')
THEN sum(qty * cost)

```

```

ELSE sum(qty * it.unit_cost)
END
,valuemat = CASE
WHEN (it.cost_method = 'A')
THEN sum(qty * m.matl_cost)
ELSE sum(qty * it.matl_cost)
END
,valuelab = CASE
WHEN (it.cost_method = 'A')
THEN sum(qty * m.lbr_cost)
ELSE sum(qty * it.lbr_cost)
END
,valuevov = CASE
WHEN (it.cost_method = 'A')
THEN sum(qty * m.vovhd_cost)
ELSE sum(qty * it.vovhd_cost)
END
INTO #tran
FROM matltran_mst m
INNER JOIN item_mst it ON m.item = it.item
WHERE trans_date >= @invdte
AND m.trans_type <> 'C'
AND m.loc <> '20'
GROUP BY m.item
,it.cost_method
,m.whse

SELECT c.item
,c.whse
,c.description
,c.ufprofile
,c.qty_rsvd -- Added by Ken 8/23/2022
,c.qty_on_hand - isnull(t.qty_on_hand,0) as Qty
,(c.qty_on_hand * i.unit_weight) - isnull((t.qty_on_hand * i.unit_weight),0) as
Lbs
,c.value - isnull(t.value,0) as Value
,(c.value - isnull(t.value,0))/Nullif(((c.qty_on_hand - isnull(t.qty_on_hand,0)) *
i.unit_weight),0) '$/Lb'
,c.valuemat - isnull(t.valuemat,0) as matvalue
,c.valuelab - isnull(t.valuelab,0) as labvalue
,c.valuevov - isnull(t.valuevov,0) as vovvalue
FROM #current c
LEFT JOIN #tran t ON c.item = t.item and c.whse = t.whse
INNER JOIN item_mst i ON c.item = i.item
WHERE (
c.qty_on_hand <> isnull(t.qty_on_hand, 0.0)
OR c.value <> isnull(t.value, 0.0)
)
AND c.item NOT LIKE 'Pfrt%'

DROP TABLE #current
DROP TABLE #tran
DROP TABLE #railloc

```

7

```
FROM [Synteline9_App].[dbo].[lot_loc_mst]
inner join lot_mst on lot_loc_mst.lot = lot_mst.lot
inner join [item_mst] on [item_mst].[item] = lot_mst.item
inner join [location_mst] on [location_mst].[loc] =
[lot_loc_mst].[loc]
where [lot_loc_mst].[loc] like 'WHS%'
order by whse asc
```

```
select _Account,
_Description,
_entity,
_Source,
_location,
Replace(_CustomerOrVendorNumber, ',', ' ')
from [dbo].[TNG_VenaGL]
where _Account = 20100 and _CustomerOrVendorNumber like '%,%'
```

[GreatTechRivalry_ChinavsUS_211207.pdf \(belfercenter.org\)](#)

[BYD EV SEDEMA.pdf \(theicct.org\)](#)

LMS Access

Your LMS is now accessible via the following link:

<https://tangentmaterials.convergencetraining.com/TangentTraining>. I have added Dan, Kevin, and Ken as users, and assigned each the Master security role. As we discussed, your usernames are your firstname.lastname (not case sensitive), and your initial password is Password1 (case sensitive). It should force you to set your password on your first login. (If for some reason it doesn't, you can change your password anytime you like from the Home tab, Dashboard, Change My Password.) (kwwass@EmilyWu19c1)

```
USE [Synteline9_App_Pilot]
GO
```

```
/***** Object: View [dbo].[TNG_VenaGL] Script Date: 8/19/2022 2:41:41 PM *****/
SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER ON
GO
```

```
ALTER view [dbo].[TNG_VenaGL] as
```

```
select gl.acct as _Account,
chart.description as _Description,
gl.site_ref as _entity,
'Synteline' as _Source,
acct_unit1 as _location,
unit1.description as _locationDescription,
acct_unit2 as _Unit2,
unit2.description As _Unit2Description,
acct_unit3 as _Unit3,
unit3.description As _Unit3Description,
acct_unit4 as _Unit4,
unit4.description As _Unit4Description,
control_year as _Year,
control_period as _period,
'Actual' as _scenario,
gl.curr_code as _Currency,
dom_amount as _Value,
ref as _Reference,
trans_date as _transDate,
from_id as _PostedFrom,
gl.vend_num as _VendorOrCustomerNumber,
case when ven.name is null then cus.name else ven.name end as _CustomerOrVendorNumber,
trans_num as _Sequence
```

```
FROM [Synteline9_App_Pilot].[dbo].[ledger_mst] gl
inner join chart_mst chart on chart.acct=gl.acct
left join vendaddr_mst ven on ven.vend_num=gl.vend_num
left join custaddr_mst cus on cus.cust_num=gl.vend_num and cus.cust_seq=0
left join unitcd1_mst unit1 on unit1.unit1=gl.acct_unit1
left join unitcd2_mst unit2 on unit2.unit2=gl.acct_unit2
left join unitcd3_mst unit3 on unit3.unit3=gl.acct_unit3
left join unitcd4_mst unit4 on unit4.unit4=gl.acct_unit4
-- where trans_date>'2019-12-31 23:59:59'
where trans_date >= (SELECT DATEADD(DAY, 1, EOMONTH(GETDATE(), -1))) and
trans_date <= (SELECT EOMONTH (GETDATE()))
```

```
Union
```

```
select gl.acct as _Account,
chart.description as _Description,
gl.site_ref as _entity,
'Synteline' as _Source,
acct_unit1 as _location,
```




```
unit1.description as _locationDescription,
acct_unit2 as _Unit2,
unit2.description As _Unit2Description,
acct_unit3 as _Unit3 ,
unit3.description As _Unit3Description,
acct_unit4 as _Unit4,
unit4.description As _Unit4Description,
control_year as _Year,
control_period as _period,
'Actual' as _scenario,
gl.curr_code as _Currency,
dom_amount as _Value,
ref as _Reference,
trans_date as _transDate,
from_id as _PostedFrom,
gl.vend_num as _VendorOrCustomerNumber,
case when ven.name is null then cus.name else ven.name end as _CustomerOrVendorNumber,
trans_num as _Sequence
```

```
FROM [AcqS_App_Pilot].[dbo].[ledger_mst] gl
inner join chart_mst chart on chart.acct=gl.acct
left join vendaddr_mst ven on ven.vend_num=gl.vend_num
left join custaddr_mst cus on cus.cust_num=gl.vend_num and cus.cust_seq=0
left join unitcd1_mst unit1 on unit1.unit1=gl.acct_unit1
left join unitcd2_mst unit2 on unit2.unit2=gl.acct_unit2
left join unitcd3_mst unit3 on unit3.unit3=gl.acct_unit3
left join unitcd4_mst unit4 on unit4.unit4=gl.acct_unit4
-- where trans_date>'2019-12-31 23:59:59'
where trans_date >= (SELECT DATEADD(DAY, 1, EOMONTH(GETDATE(), -1))) and
trans_date <= (SELECT EOMONTH (GETDATE()))
```

Union

```
select gl.acct as _Account,
chart.description as _Description,
gl.site_ref as _entity,
'Syteline' as _Source,
acct_unit1 as _location,
unit1.description as _locationDescription,
acct_unit2 as _Unit2,
unit2.description As _Unit2Description,
acct_unit3 as _Unit3 ,
unit3.description As _Unit3Description,
acct_unit4 as _Unit4,
unit4.description As _Unit4Description,
control_year as _Year,
control_period as _period,
'Actual' as _scenario,
gl.curr_code as _Currency,
dom_amount as _Value,
ref as _Reference,
trans_date as _transDate,
from_id as _PostedFrom,
gl.vend_num as _VendorOrCustomerNumber,
case when ven.name is null then cus.name else ven.name end as _CustomerOrVendorNumber,
trans_num as _Sequence
```

```
FROM [HoldS_App_Pilot].[dbo].[ledger_mst] gl
inner join chart_mst chart on chart.acct=gl.acct
left join vendaddr_mst ven on ven.vend_num=gl.vend_num
left join custaddr_mst cus on cus.cust_num=gl.vend_num and cus.cust_seq=0
left join unitcd1_mst unit1 on unit1.unit1=gl.acct_unit1
left join unitcd2_mst unit2 on unit2.unit2=gl.acct_unit2
left join unitcd3_mst unit3 on unit3.unit3=gl.acct_unit3
left join unitcd4_mst unit4 on unit4.unit4=gl.acct_unit4
-- where trans_date>'2019-12-31 23:59:59'
where trans_date >= (SELECT DATEADD(DAY, 1, EOMONTH(GETDATE(), -1))) and trans_date
<= (SELECT EOMONTH (GETDATE()))
```

Union

```
select gl.acct as _Account,
chart.description as _Description,
gl.site_ref as _entity,
'Syteline' as _Source,
acct_unit1 as _location,
unit1.description as _locationDescription,
acct_unit2 as _Unit2,
unit2.description As _Unit2Description,
acct_unit3 as _Unit3 ,
unit3.description As _Unit3Description,
acct_unit4 as _Unit4,
unit4.description As _Unit4Description,
control_year as _Year,
control_period as _period,
'Actual' as _scenario,
gl.curr_code as _Currency,
dom_amount as _Value,
ref as _Reference,
trans_date as _transDate,
from_id as _PostedFrom,
gl.vend_num as _VendorOrCustomerNumber,
case when ven.name is null then cus.name else ven.name end as _CustomerOrVendorNumber,
trans_num as _Sequence
```

```
FROM [InvS_App_Pilot].[dbo].[ledger_mst] gl
inner join chart_mst chart on chart.acct=gl.acct
left join vendaddr_mst ven on ven.vend_num=gl.vend_num
left join custaddr_mst cus on cus.cust_num=gl.vend_num and cus.cust_seq=0
left join unitcd1_mst unit1 on unit1.unit1=gl.acct_unit1
left join unitcd2_mst unit2 on unit2.unit2=gl.acct_unit2
left join unitcd3_mst unit3 on unit3.unit3=gl.acct_unit3
left join unitcd4_mst unit4 on unit4.unit4=gl.acct_unit4
-- where trans_date>'2019-12-31 23:59:59'
where trans_date >= (SELECT DATEADD(DAY, 1, EOMONTH(GETDATE(), -1))) and trans_date
<= (SELECT EOMONTH (GETDATE()))
GO
```