Table of Contents

ITEMS 1

To Find Duplicate Item Category Code 1

Query to find on Hand Quantity 2

To Check Item Catogry For Inventory master (No Of Segments May Varry) 2

Check Locators for inventory Inventory Org Wise(Number of segment may varry) 2

To Select Unit Of measure exist in ebusiness suite 3

Query to find Item Code, Item Description Oracle Item Master Query 3

Query to Find out On Hand Quantity of specific Item Oracle inventory 3

Display all categories that the Item Belongs 4

Another Query to Get Onhand Qty With Oranization ID, Item Code, Quantity 4

Provide Month by Month Inventory 5

PURCHASING 6

Get Number Of canceled requisition 6

Query to find out info from PR to PO 6

PAYABLES 7

AP invoice distributions all to a GL Header via SLA 7

Invoices on hold 8

FIXED ASSETS 9

ORDER MANAGEMENT 11

Number of line processed in Order Management 11

Qunatity on order, Expected Deliver 11

Price List changes 12

price List /Qualifiers/Modifiers/Conext/Segments 12

CUSTOMER INFORMATION 13

Query to find out Customer Master Information. Customer Name, Account Number, Adress etc. 13

list all customers 13

ACCOUNT RECEIVABLES 14

Invoice net amt, and VAT 14

Extract revenue distribution lines in AR 14

AR Receipt SQL Queries 15

OPM 18

Formula Base Tables and Detail of Products and Ingredients 18

Query to Find out Quantity of Products and Ingredients of all Batches in particular Month. 18

USER RESPONSABILITIES 19

Query to Find Responsibilities assigned to particular user. 19

OTHER 19

Operating Unit name based on the Operating Unit ID 19

## ITEMS

### To Find Duplicate Item Category Code

SELECT category\_set\_name, category\_concat\_segments, COUNT (\*)

FROM mtl\_category\_set\_valid\_cats\_v

WHERE (category\_set\_id = 1)

GROUP BY category\_set\_name, category\_concat\_segments

HAVING COUNT (\*) > 1

ORDER BY category\_concat\_segments

### Query to find on Hand Quantity

select sum(transaction\_quantity) from MTL\_ONHAND\_QUANTITIES

where inventory\_item\_id=9

and organization\_id=188

### To Check Item Catogry For Inventory master (No Of Segments May Varry)

SELECT ood.organization\_name,

segment1|| ‘-’|| segment2|| ‘-’|| segment3 catgory

FROM org\_organization\_definitions ood,

mtl\_categories\_vl mcv,

mtl\_category\_sets mcs

WHERE mcs.structure\_id = mcv.structure\_id

ORDER BY ood.organization\_name

### Check Locators for inventory Inventory Org Wise(Number of segment may varry)

SELECT mil.segment1 loc\_seg1, mil.segment11 loc\_seg11, mil.segment2 loc\_seg2,

mil.segment3 loc\_seg3, mil.segment4 loc\_seg4, mil.segment5 loc\_seg5,

mil.segment6 loc\_seg6,ood.ORGANIZATION\_NAME,mil.SUBINVENTORY\_CODE

FROM mtl\_item\_locations mil,org\_organization\_definitions ood

where mil.ORGANIZATION\_ID = ood.ORGANIZATION\_ID

Display All Subinventories Setup

select msi.secondary\_inventory\_name, MSI.SECONDARY\_INVENTORY\_NAME “Subinventory”, MSI.DESCRIPTION “Description”,

MSI.DISABLE\_DATE “Disable Date”, msi.PICKING\_ORDER “Picking Order”,

gcc1.concatenated\_segments “Material Account”,

gcc2.concatenated\_segments “Material Overhead Account”,

gcc3.concatenated\_segments “Resource Account”,

gcc4.concatenated\_segments “Overhead Account”,

gcc5.concatenated\_segments “Outside Processing Account”,

gcc6.concatenated\_segments “Expense Account”,

gcc7.concatenated\_segments “Encumbrance Account”,

msi.material\_overhead\_account,

msi.resource\_account,

msi.overhead\_account,

msi.outside\_processing\_account,

msi.expense\_account,

msi.encumbrance\_account

from mtl\_secondary\_inventories msi,

gl\_code\_combinations\_kfv gcc1,

gl\_code\_combinations\_kfv gcc2,

gl\_code\_combinations\_kfv gcc3,

gl\_code\_combinations\_kfv gcc4,

gl\_code\_combinations\_kfv gcc5,

gl\_code\_combinations\_kfv gcc6,

gl\_code\_combinations\_kfv gcc7

where msi.material\_account = gcc1.CODE\_COMBINATION\_ID(+)

and msi.material\_overhead\_account = gcc2.CODE\_COMBINATION\_ID(+)

and msi.resource\_account = gcc3.CODE\_COMBINATION\_ID(+)

and msi.overhead\_account = gcc4.CODE\_COMBINATION\_ID(+)

and msi.outside\_processing\_account = gcc5.CODE\_COMBINATION\_ID(+)

and msi.expense\_account = gcc6.CODE\_COMBINATION\_ID(+)

and msi.encumbrance\_account = gcc7.CODE\_COMBINATION\_ID(+)

order by msi.secondary\_inventory\_name

### To Select Unit Of measure exist in ebusiness suite

select uom\_code,unit\_of\_measure

from mtl\_units\_of\_measure

### Query to find Item Code, Item Description Oracle Item Master Query

(select item, description from mtl\_system\_items\_b

where inventory\_item\_id=&your\_item

and organization\_id=&organization\_id) item,

### Query to Find out On Hand Quantity of specific Item Oracle inventory

(select sum(transaction\_quantity) from mtl\_onhand\_quantity\_details

where inventory\_item\_id=&your\_item

and organization\_id=&organization\_id) onhand,

– Qty Issued by X No clue what you want here

–Qty On Order,Expected deivery date(select sum(ordered\_quantity),scheduled\_ship\_date from oe\_order\_lines\_all

where inventory\_item\_id=&your\_item

and ship\_from\_org\_id=&organization\_id

group by scheduled\_ship\_date) order\_info,

–Toatl Received Qty

(select sum(transaction\_quantity) from mtl\_material\_transactions

inventory\_item\_id=&your\_item

and organization\_id=&organization\_id

and transaction\_quantity>0) tot\_rec\_qty,

–Total received Qty in 9 months

(select sum(transaction\_quantity) from mtl\_material\_transactions

inventory\_item\_id=&your\_item

and organization\_id=&organization\_id

and transaction\_quantity>0

and transaction\_date between trunc(sysdate) and trunc(sysdate-270)) tot\_rec\_qty\_9mths,

–Total issued quantity in 9 months(select sum(transaction\_quantity) from mtl\_material\_transactions

inventory\_item\_id=&your\_item

and organization\_id=&organization\_id

and transaction\_quantity<0 and transaction\_date between trunc(sysdate) and trunc(sysdate-270)) tot\_iss\_qty\_9mths, –Average monthly consumption

(select sum(transaction\_quantity)/30 from mtl\_material\_transactions

inventory\_item\_id=&your\_item

and organization\_id=&organization\_id

and transaction\_quantity<0) avg\_month\_consumption; Find detail of specific Applications

### Display all categories that the Item Belongs

SELECTunique micv.CATEGORY\_SET\_NAME “Category Set”,

micv.CATEGORY\_SET\_ID “Category Set ID”,

decode( micv.CONTROL\_LEVEL,

1, ‘Master’,

2, ‘Org’,

‘Other’) “Control Level”,

micv.CATEGORY\_ID “Category ID”,

micv.CATEGORY\_CONCAT\_SEGS “Category”

FROM

MTL\_ITEM\_CATEGORIES\_V micv

### Another Query to Get Onhand Qty With Oranization ID, Item Code, Quantity

SELECT organization\_id,

(SELECT ( msib.segment1|| ‘-’|| msib.segment2|| ‘-’|| msib.segment3|| ‘-’|| msib.segment4)

FROM mtl\_system\_items\_b msib

WHERE msib.inventory\_item\_id = moq.inventory\_item\_id

AND msib.organization\_id = moq.organization\_id) “Item Code”,

(SELECT description

FROM mtl\_system\_items\_b msib

WHERE msib.inventory\_item\_id =

moq.inventory\_item\_id

AND msib.organization\_id = moq.organization\_id)

“item Description”,

SUM (moq.transaction\_quantity) onhandqty

FROM mtl\_onhand\_quantities moq

GROUP BY moq.organization\_id, (moq.inventory\_item\_id)

### Provide Month by Month Inventory

SELECT to\_char(TRANSACTION\_DATE,'YYYY-MM'),mp.organization\_code, msi.inventory\_item\_id, msi.segment1 item\_code,

msi.description,

/\* mmt.subinventory\_code, mil.segment1 row1,

mil.segment2 rack, mil.segment3 bin, \*/

SUM (mmt.primary\_quantity) primary\_quantity,

DECODE (SUM (mmt.primary\_quantity),

0, 0,

SUM ((mmt.primary\_quantity) \* mmt.actual\_cost)

/ SUM (mmt.primary\_quantity)

) actual\_cost,

DECODE (SUM (mmt.primary\_quantity),

0, 0,

SUM ((mmt.primary\_quantity) \* mmt.transaction\_cost)

/ SUM (mmt.primary\_quantity)

) transaction\_cost,

DECODE (SUM (mmt.primary\_quantity),

0, 0,

SUM ((mmt.primary\_quantity) \* mmt.new\_cost)

/ SUM (mmt.primary\_quantity)

) new\_cost

FROM inv.mtl\_material\_transactions mmt,

inv.mtl\_parameters mp,

inv.mtl\_system\_items\_b msi,

inv.mtl\_item\_locations mil,

inv.mtl\_transaction\_types mtt

WHERE mmt.organization\_id = mp.organization\_id

AND mmt.organization\_id = msi.organization\_id

AND mmt.inventory\_item\_id = msi.inventory\_item\_id

AND mmt.organization\_id = mil.organization\_id(+)

AND mmt.subinventory\_code = mil.subinventory\_code(+)

AND mmt.locator\_id = mil.inventory\_location\_id(+)

AND mmt.transaction\_type\_id = mtt.transaction\_type\_id

AND msi.segment1 = NVL (:p\_item\_code, msi.segment1)

AND mp.organization\_code = NVL (:p\_warehouse\_code, mp.organization\_code)

AND mmt.transaction\_date < TO\_DATE (:p\_as\_on\_dt, 'DD-MON-YYYY')+1

AND mtt.transaction\_type\_name NOT IN (

'Account issue',

'Account receipt',

'Account alias issue',

'Account alias receipt',

'Cycle Count Adjust',

'Cycle Count Transfer',

'Logical RMA Receipt',

'COGS Recognition',

'Logical Sales Order Issue',

-- 'Sales Order Pick',

'Standard cost update'

)

GROUP BY to\_char(TRANSACTION\_DATE,'YYYY-MM'),

msi.inventory\_item\_id,

msi.segment1,

msi.description,

mp.organization\_code

/\* , mmt.subinventory\_code,

mil.segment1,

mil.segment2,

mil.segment3 \*/

HAVING SUM(mmt.primary\_quantity) >0

order by to\_char(TRANSACTION\_DATE,'YYYY-MM'),

msi.inventory\_item\_id,

msi.segment1,

msi.description,

mp.organization\_code

/\* ,mmt.subinventory\_code,

mil.segment1,

mil.segment2,

mil.segment3 \*/

## PURCHASING

### Get Number Of canceled requisition

SELECT a.AUTHORIZATION\_STATUS,(a.ORG\_ID),(SELECT distinct hr.per\_all\_people\_f.first\_name|| ‘ ‘|| hr.per\_all\_people\_f.middle\_names|| ‘ ‘|| hr.per\_all\_people\_f.last\_name “Employee Name”

FROM hr.per\_all\_people\_f

where hr.per\_all\_people\_f.PERSON\_ID in

(select employee\_id from fnd\_user fu where fu.user\_id = a.CREATED\_BY)) CREATED\_BY,count(SEGMENT1 )

FROM

po\_requisition\_headers\_all a

WHERE

a.creation\_date BETWEEN TO\_DATE(’01/01/2007′, ‘DD/MM/YYYY’)

and TO\_DATE(’30/05/2007′, ‘DD/MM/YYYY’)

and a.AUTHORIZATION\_STATUS = ‘CANCELLED’

group by a.AUTHORIZATION\_STATUS,a.ORG\_ID,a.CREATED\_BY

### Query to find out info from PR to PO

select distinct u.description “Requestor”,

porh.segment1 as “Req Number”,

trunc(porh.Creation\_Date) “Created On”,

pord.LAST\_UPDATED\_BY,

porh.Authorization\_Status “Status”,

porh.Description “Description”,

poh.segment1 “PO Number”,

trunc(poh.Creation\_date) “PO Creation Date”,

poh.AUTHORIZATION\_STATUS “PO Status”,

trunc(poh.Approved\_Date) “Approved Date”

from apps.po\_headers\_all poh,

apps.po\_distributions\_all pod,

apps.po\_req\_distributions\_all pord,

apps.po\_requisition\_lines\_all porl,

apps.po\_requisition\_headers\_all porh,

apps.fnd\_user u

where porh.requisition\_header\_id = porl.requisition\_header\_id

and porl.requisition\_line\_id = pord.requisition\_line\_id

and pord.distribution\_id = pod.req\_distribution\_id(+)

and pod.po\_header\_id = poh.po\_header\_id(+)

and porh.created\_by = u.user\_id

order by 2

## PAYABLES

### AP invoice distributions all to a GL Header via SLA

SELECT c.code\_combination\_id, h.je\_header\_id, l.ae\_header\_id, l.ae\_line\_num, te.source\_id\_int\_1, te.application\_id, te.entity\_id,

h.je\_source,

h.je\_category,

i.gl\_date,

s.vendor\_name,

s.segment1 as supplier\_no,

l.event\_class\_code as event\_class, i.invoice\_id, ad.invoice\_distribution\_id,

i.invoice\_num AS transaction\_number,

i.invoice\_date,

initcap(jl.description) description,

jl.accounted\_dr as debit,

jl.accounted\_cr as credit,

Nvl(Jl.Accounted\_Dr, 0) – Nvl(Jl.Accounted\_Cr, 0) Net\_Amount

From Apps.Gl\_Je\_Headers H,

apps.gl\_je\_lines jl,

Apps.Gl\_Code\_Combinations C,

apps.gl\_import\_references r,

apps.xla\_ae\_lines al,

Apps.Xla\_Ae\_headers Ah,

apps.xla\_distribution\_links l,

apps.ap\_invoices\_all i,

apps.ap\_invoice\_distributions\_all ad,

apps.ap\_suppliers s,

apps.xla\_events e,

apps.xla\_transaction\_entities te

where ad.accounting\_date between :startdate and :enddate

and c.code\_combination\_id = 6429

and jl.description != ‘GB VAT – STANDARD TAX’

and ad.line\_type\_lookup\_code = ‘ITEM’

AND jl.je\_header\_id = h.je\_header\_id

AND jl.code\_combination\_id = c.code\_combination\_id

and al.gl\_sl\_link\_id = r.gl\_sl\_link\_id

and al.ae\_header\_id = ah.ae\_header\_id

and al.application\_id = ah.application\_id

and ah.application\_id = e.application\_id

and ah.event\_id = e.event\_id

and e.application\_id = te.application\_id(+)

and e.entity\_id = te.entity\_id(+)

AND r.je\_header\_id = jl.je\_header\_id

AND r.je\_line\_num = jl.je\_line\_num

AND l.ae\_header\_id = al.ae\_header\_id

and l.ae\_line\_num = al.ae\_line\_num

and l.applied\_to\_source\_id\_num\_1 = i.invoice\_id

and l.source\_distribution\_id\_num\_1 = ad.invoice\_distribution\_id

and ad.invoice\_id = i.invoice\_id

and i.vendor\_id = s.vendor\_id

order by i.gl\_date desc

### Invoices on hold

select distinct

ai.invoice\_num,ai.invoice\_amount,ai.invoice\_date

,ai.invoice\_received\_date,ah.hold\_lookup\_code,ah.hold\_reason,pv.vendor\_name,pv.segment1 vendor\_number

,ph.segment1 po\_number

,gcc.segment1||’.’||gcc.segment2||’.’||gcc.segment3||’.’||gcc.segment4||’.’||gcc.segment5||’.’||gcc.segment6

–,aid.AMOUNT

from ap\_invoices\_all ai

,ap\_holds\_all ah

,po\_vendors pv

,po\_headers\_all ph

,ap\_invoice\_distributions\_all aid

,po\_distributions\_all pda

,gl\_code\_combinations gcc

where 1=1

and ai.invoice\_id = ah.invoice\_id

and ai.vendor\_id = pv.vendor\_id

–and ai.po\_header\_id = ph.po\_header\_id(+)

and ph.PO\_HEADER\_ID = pda.PO\_HEADER\_ID

and ah.invoice\_id = aid.invoice\_id

and aid.po\_distribution\_id = pda.po\_distribution\_id

and aid.dist\_code\_combination\_id = gcc.code\_combination\_id

### FIXED ASSETS

SELECT FCB.category\_id, fcb.SEGMENT1 “Major”, fcb.SEGMENT2 “Minor”,fct.DESCRIPTION   
FROM APPS.FA\_CATEGORIES\_b FCB, APPS.FA\_CATEGORIES\_TL FCT  
WHERE FCT.CATEGORY\_ID=FCB.CATEGORY\_ID  
AND FCB.ENABLED\_FLAG=’Y’

Or use this:

SELECT parent\_flex\_value\_low “Major”,flex\_value “Minor”, enabled\_flag, start\_date\_active, end\_date\_active  
FROM fnd\_flex\_value\_sets ffvs  
, fnd\_flex\_values\_vl ffv  
WHERE ffvs.flex\_value\_set\_id = ffv.flex\_value\_set\_id  
AND ffvs.flex\_value\_set\_name =  
and enabled\_flaG=’Y’  
AND SYSDATE BETWEEN NVL(START\_DATE\_ACTIVE, SYSDATE-1) AND NVL(END\_dATE\_aCTIVE,SYSDATE+1)

select a.book\_type\_code,

a.asset\_id, b.asset\_number,c.description, a.date\_placed\_in\_service, a.deprn\_start\_date,a.deprn\_method\_code,a.original\_ cost,

a.adjusted\_cost, a.cost, a.adjustment\_required\_status, a.capitalize\_flag, a.basic\_rate, a.adjusted\_rate,

d.deprn\_amount, d.ytd\_deprn,e.fiscal\_year, e.period\_num

from fa\_books a, fa\_additions\_b b, fa\_additions\_tl c ,

fa\_deprn\_summary d,

fa\_deprn\_periods e

where a.asset\_id=b.asset\_id

and a.asset\_id=c.asset\_id

and a.asset\_id= d.asset\_id

and d.period\_counter=e.period\_counter

and d.deprn\_reserve <> 0

order by a.book\_type\_code,e.fiscal\_year, e.period\_num;

select

fp.period\_name, dhcc.segment2 business\_unit, dhcc.segment4 dept, dhcc.segment3

account, adt.asset\_number,

adt.tag\_number, ltrim(rtrim(cat.segment1))||'-'||ltrim(rtrim(cat.segment2))

||'-'||ltrim(rtrim(cat.segment3)) category,

bks.date\_placed\_in\_service acquistion\_date, bks.original\_cost, adt.

description, dh.location\_id, adt.context subject\_to\_property\_tax,

adt.attribute1

property\_tax\_code, round(sum(decode(bks.period\_counter\_fully\_retired, '',bks

.cost, 0)

\* dh.units\_assigned / ah.units), 2) COST, round(sum(nvl(dn.deprn\_amount,0) \*

dh.units\_assigned/ ah.units), 2) deprn,

round(sum(nvl(dn.deprn\_reserve,0) \* dh.units\_assigned/ ah.units),

2)deprn\_reserve

, round(sum(nvl(dn.ytd\_deprn,0)

\* dh.units\_assigned/ ah.units), 2) YTD\_DEP, round(sum(decode(bks.

period\_counter\_fully\_retired, '', (bks.cost - dn.deprn\_reserve), 0) \* dh.

units\_assigned/ ah.units), 2) c\_nbv

from fa\_distribution\_history dh, fa\_asset\_history ah, fa\_additions

adt,fa\_categories\_b cat

, fa\_books bks, gl\_code\_combinations dhcc, fa\_deprn\_summary

dn,fa\_deprn\_periods fp

where fp.book\_type\_code = 'XXX'

and dn.book\_type\_code = 'XXXX'

and dn.period\_counter = ( select dp.period\_counter

from fa\_deprn\_periods dp

where dp.book\_type\_code = 'XXXX'

and dp.period\_counter

( select max(dpz.period\_counter)

from fa\_deprn\_summary dsz,

fa\_deprn\_periods dpz

where dpz.book\_type\_code = 'XXXX'

and dpz.period\_counter <= fp.period\_counter

and dsz.book\_type\_code = 'XXXX'

and dsz.period\_counter = dpz.period\_counter

and dsz.asset\_id = dn.asset\_id ))

and bks.book\_type\_code = 'XXXX'

and bks.asset\_id = dn.asset\_id

and nvl(bks.date\_ineffective,sysdate) >

to\_date(to\_char(nvl(fp.period\_close\_date, sysdate),'DD-MM-YYYY

HH24:MI:SS'),'DD-MM-YYYY

HH24:MI:SS')

and bks.date\_effective <

to\_date(to\_char(nvl(fp.period\_close\_date, sysdate),'DD-MM-YYYY

HH24:MI:SS'),'DD-MM-YYYY

HH24:MI:SS')

and nvl(bks.period\_counter\_fully\_retired, fp.period\_counter) in

( select dpy.period\_counter

from fa\_deprn\_periods dpy

where dpy.book\_type\_code = 'XXXX'

and dpy.fiscal\_year = fp.fiscal\_year)

and adt.asset\_id = dn.asset\_id

and adt.ASSET\_CATEGORY\_ID = cat.category\_id

and adt.asset\_id = dh.asset\_id

and dh.book\_type\_code = 'XXXX'

and nvl(dh.date\_ineffective,sysdate) >

to\_date(to\_char(nvl(fp.period\_close\_date, sysdate),'DD-MM-YYYY

HH24:MI:SS'),'DD-MM-YYYY

HH24:MI:SS')

and dh.date\_effective <

to\_date(to\_char(nvl(fp.period\_close\_date, sysdate),'DD-MM-YYYY

HH24:MI:SS'),'DD-MM-YYYY

HH24:MI:SS')

and dhcc.code\_combination\_id(+) = dh.code\_combination\_id

and ah.asset\_id = adt.asset\_id

and nvl(ah.date\_ineffective,sysdate) >

to\_date(to\_char(nvl(fp.period\_close\_date, sysdate),'DD-MM-YYYY

HH24:MI:SS'),'DD-MM-YYYY

HH24:MI:SS')

and ah.date\_effective <

to\_date(to\_char(nvl(fp.period\_close\_date, sysdate),'DD-MM-YYYY

HH24:MI:SS'),'DD-MM-YYYY

HH24:MI:SS')

group by

fp.period\_name,

dhcc.segment2,

dhcc.segment4,

dhcc.segment3,

adt.asset\_number,

adt.tag\_number,

ltrim(rtrim(cat.segment1))||'-'||ltrim(rtrim(cat.segment2))||'-'||

ltrim(rtrim(cat.segment3)),

adt.description,

bks.date\_placed\_in\_service,

bks.original\_cost,

dh.location\_id,

adt.context,

adt.attribute1;

## ORDER MANAGEMENT

### Number of line processed in Order Management

SELECT COUNT (line\_id) “Order Line Processed”

FROM oe\_order\_lines\_all

WHERE creation\_date BETWEEN TO\_DATE (:Fdate, ‘DD/MM/YYYY’)

AND TO\_DATE (:tdate, ‘DD/MM/YYYY’)

AND flow\_status\_code = ‘CLOSED’;

### Qunatity on order, Expected Deliver

select

sum(ordered\_quantity),a.SCHEDULE\_SHIP\_DATE

from oe\_order\_lines\_all a

where inventory\_item\_id=10

and ship\_from\_org\_id=188

group by a.SCHEDULE\_SHIP\_DATE

### Price List changes

select c.ordered\_item, c.unit\_list\_price, c.unit\_selling\_price, a.operand

from qp\_list\_lines a,

qp\_pricing\_attributes b,

oe\_order\_lines\_all c

where a.list\_line\_id = b.list\_line\_id

and a.list\_line\_type\_code = 'PLL'

and b.product\_attribute\_context = 'ITEM'

and b.product\_attribute = 'PRICING\_ATTRIBUTE1'

and c.inventory\_item\_id = b.product\_attr\_value

and c.pricing\_quantity\_uom = b.product\_uom\_code

and a.arithmetic\_operator = 'UNIT\_PRICE'

and c.header\_id = :header\_id

and c.price\_list\_id = a.list\_header\_id

and c.pricing\_date between nvl(a.start\_date\_active,c.pricing\_date)

and nvl(a.end\_date\_active,c.pricing\_date)

### price List /Qualifiers/Modifiers/Conext/Segments

SELECT l.list\_line\_id,q.qualifier\_grouping\_no,

q.qualifier\_id, q.qualifier\_context, q.qualifier\_attr\_value,

q.comparison\_operator\_code,q.qualifier\_precedence,q.qual\_attr\_value\_from\_number,

q.qualifier\_attribute,q.end\_date\_active,l.end\_date\_active,h.end\_date\_active

FROM

qp\_list\_headers\_all h,

qp\_list\_lines l,

qp\_qualifiers q

where h.list\_header\_id = l.list\_header\_id

and h.list\_header\_id = q.list\_header\_id

and h.list\_header\_id = &list\_id -- Price List Header ID or Modifier header ID

and NVL(h.end\_date\_active,sysdate) >= sysdate

and NVL(l.end\_date\_active,sysdate) >= sysdate

and NVL(q.end\_date\_active,sysdate) >= sysdate;

select q.qualifier\_id,q.qualifier\_context,q.qualifier\_attribute,qualifier\_attr\_value ,

ct.prc\_context\_id, qs.segment\_code

from qp\_qualifiers q , qp\_prc\_contexts\_b ct , qp\_segments\_b qs

where q.list\_header\_id = &ListHeaderID --PriceList Header ID

and ct.prc\_context\_type ='QUALIFIER'

and q.qualifier\_context = ct.prc\_context\_code

and qs.prc\_context\_id = ct.prc\_context\_id

and qs.segment\_mapping\_column = q.qualifier\_attribute

## CUSTOMER INFORMATION

### Query to find out Customer Master Information. Customer Name, Account Number, Adress etc.

select p.PARTY\_NAME,ca.ACCOUNT\_NUMBER,loc.address1,loc.address2,loc.address3,loc.city,loc.postal\_code,

loc.country,ca.CUST\_ACCOUNT\_ID

from apps.ra\_customer\_trx\_all I,

apps.hz\_cust\_accounts CA,

apps.hz\_parties P,

apps.hz\_locations Loc,

apps.hz\_cust\_site\_uses\_all CSU,

apps.hz\_cust\_acct\_sites\_all CAS,

apps.hz\_party\_sites PS

where I.COMPLETE\_FLAG =’Y’

and I.bill\_TO\_CUSTOMER\_ID= CA.CUST\_ACCOUNT\_ID

and ca.PARTY\_ID=p.PARTY\_ID

and I.bill\_to\_site\_use\_id=csu.site\_use\_id

and csu.CUST\_ACCT\_SITE\_ID=cas.CUST\_ACCT\_SITE\_ID

and cas.PARTY\_SITE\_ID=ps.party\_site\_id

and ps.location\_id=loc.LOCATION\_ID

### list all customers

SELECT arps.org\_id, hzp.party\_name customer\_name,

hca.account\_number customer\_number,

-- rat.NAME term\_name,

arps.invoice\_currency\_code,

SUM (arps.amount\_due\_remaining) outstandaing\_amt

FROM --apps.hz\_cust\_profile\_amts hcpa,

-- apps.hz\_customer\_profiles hcp,

apps.hz\_cust\_accounts\_all hca,

apps.hz\_cust\_acct\_sites\_all hcas,

apps.hz\_cust\_site\_uses\_all hcsu,

apps.hz\_parties hzp,

-- ra\_terms\_tl rat,

ar\_payment\_schedules\_all arps

WHERE 1 = 1

--AND hca.cust\_account\_id = hcp.cust\_account\_id

--AND hca.cust\_account\_id = hcpa.cust\_account\_id

AND hca.cust\_account\_id = hcas.cust\_account\_id

AND hcas.cust\_acct\_site\_id = hcsu.cust\_acct\_site\_id

AND hcsu.site\_use\_code = 'BILL\_TO'

AND hcsu.status = 'A'

--AND hcpa.site\_use\_id IS NULL

--AND hcp.site\_use\_id IS NULL

--AND hcpa.overall\_credit\_limit <> 0

AND arps.payment\_schedule\_id > 0

AND arps.amount\_due\_remaining <> 0

AND hzp.party\_id = hca.party\_id

--AND hcsu.payment\_term\_id = rat.term\_id

AND hca.cust\_account\_id = arps.customer\_id

AND arps.status = 'OP'

GROUP BY hca.account\_number,

hzp.party\_name,

arps.invoice\_currency\_code,

-- rat.NAME,

arps.org\_id

-- hcpa.overall\_credit\_limit

## ACCOUNT RECEIVABLES

### Invoice net amt, and VAT

SELECT party\_name, account\_number, rct.trx\_number,

(SELECT SUM (extended\_amount)

FROM ra\_customer\_trx\_lines\_all

WHERE customer\_trx\_id = rct.customer\_trx\_id

AND line\_type = 'LINE') net\_amount,

(SELECT SUM (extended\_amount)

FROM ra\_customer\_trx\_lines\_all

WHERE customer\_trx\_id = rct.customer\_trx\_id

AND line\_type = 'TAX') vat\_amount,

SUM (extended\_amount) gross\_amount, rctd.gl\_date

FROM hz\_parties hzp,

hz\_cust\_accounts hca,

ra\_customer\_trx\_all rct,

ra\_customer\_trx\_lines\_all rctl,

ra\_cust\_trx\_line\_gl\_dist\_all rctd

WHERE hzp.party\_id = hca.party\_id

AND hca.cust\_account\_id = rct.bill\_to\_customer\_id

AND rct.customer\_trx\_id = rctl.customer\_trx\_id

AND rctl.customer\_trx\_line\_id = rctd.customer\_trx\_line\_id

GROUP BY party\_name,

account\_number,

rct.trx\_number,

rctd.gl\_date,

rct.customer\_trx\_id

### Extract revenue distribution lines in AR

SELECT distinct c.customer\_name,

c.customer\_number,

c.customer\_id,

t.customer\_trx\_id,

t.trx\_number,

ct.NAME invoice\_type,

l.line\_number,

t.org\_id,

cc.segment1,

cc.segment2,

cc.segment3,

cc.segment4,

cc.segment5,

cc.segment6,

d.gl\_date,

d.cust\_trx\_line\_gl\_dist\_id,

d.code\_combination\_id,

d.account\_class

FROM ra\_cust\_trx\_types\_all ct,

ra\_customers c,

ra\_customer\_trx\_all t,

ra\_customer\_trx\_lines\_all l,

gl\_code\_combinations cc,

ra\_cust\_trx\_line\_gl\_dist\_all d

WHERE 1 = 1

AND t.cust\_trx\_type\_id = ct.cust\_trx\_type\_id

AND t.bill\_to\_customer\_id = c.customer\_id

AND d.customer\_trx\_id = t.customer\_trx\_id

AND d.customer\_trx\_line\_id = l.customer\_trx\_line\_id(+)

AND d.code\_combination\_id = cc.code\_combination\_id

AND TRUNC (d.gl\_date) >= TO\_DATE (’01-01-2009′, ‘DD-MM-YYYY’)

AND d.posting\_control\_id = -3

AND d.account\_set\_flag = ‘N’

AND d.account\_class = ‘REV’

### AR Receipt SQL Queries

-- Query for Total On Account Receipt Amount

SELECT NVL(SUM(ps.amount\_due\_remaining), 0) total\_onacct\_receipts

FROM hz\_cust\_accounts\_all cust\_acct,

ar\_payment\_schedules\_all ps,

ar\_receivable\_applications\_all arr,

hz\_cust\_acct\_sites\_all acct\_site,

hz\_party\_sites party\_site,

hz\_locations loc,

hz\_cust\_site\_uses\_all site\_uses,

ar\_cash\_receipts\_all acr,

ar\_cash\_receipt\_history\_all crh,

gl\_code\_combinations cc

WHERE TRUNC(ps.gl\_date) <= :p\_as\_of\_date

AND ps.customer\_id = cust\_acct.cust\_account\_id

AND cust\_acct.account\_number = :p\_account\_number

AND ps.customer\_id = cust\_acct.cust\_account\_id

AND acct\_site.party\_site\_id = party\_site.party\_site\_id

AND loc.location\_id = party\_site.location\_id

AND ps.cash\_receipt\_id = acr.cash\_receipt\_id

AND acr.cash\_receipt\_id = crh.cash\_receipt\_id

AND crh.account\_code\_combination\_id = cc.code\_combination\_id

AND ps.trx\_date <= :p\_as\_of\_date

AND ps.CLASS = 'PMT'

AND ps.cash\_receipt\_id = arr.cash\_receipt\_id

AND arr.status IN ('ACC')

AND ps.status = 'OP'

AND site\_uses.site\_use\_code = 'BILL\_TO'

AND site\_uses.cust\_acct\_site\_id = acct\_site.cust\_acct\_site\_id

AND NVL(site\_uses.status, 'A') = 'A'

AND cust\_acct.cust\_account\_id = acct\_site.cust\_account\_id

AND acct\_site.cust\_acct\_site\_id = site\_uses.cust\_acct\_site\_id

AND ps.customer\_id = acct\_site.cust\_account\_id

AND ps.customer\_site\_use\_id = site\_uses.site\_use\_id HAVING

NVL(SUM(arr.amount\_applied), 0) > 0;

-- Query for Total Unapplied Receipt Amount

SELECT NVL(SUM(arr.amount\_applied), 0) total\_unapp\_receipts

FROM hz\_cust\_accounts\_all cust\_acct,

ar\_payment\_schedules\_all ps,

ar\_receivable\_applications\_all arr,

hz\_cust\_acct\_sites\_all acct\_site,

hz\_party\_sites party\_site,

hz\_locations loc,

hz\_cust\_site\_uses\_all site\_uses,

ar\_cash\_receipts\_all acr,

ar\_cash\_receipt\_history\_all crh,

gl\_code\_combinations cc

WHERE TRUNC(ps.gl\_date) <= :p\_as\_of\_date

AND ps.customer\_id = cust\_acct.cust\_account\_id

AND cust\_acct.account\_number = :p\_account\_number

AND ps.customer\_id = cust\_acct.cust\_account\_id

AND acct\_site.party\_site\_id = party\_site.party\_site\_id

AND loc.location\_id = party\_site.location\_id

AND ps.cash\_receipt\_id = acr.cash\_receipt\_id

AND acr.cash\_receipt\_id = crh.cash\_receipt\_id

AND crh.account\_code\_combination\_id = cc.code\_combination\_id

AND ps.trx\_date <= :p\_as\_of\_date

AND ps.CLASS = 'PMT'

AND ps.cash\_receipt\_id = arr.cash\_receipt\_id

AND arr.status = 'UNAPP'

AND ps.status = 'OP'

AND site\_uses.site\_use\_code = 'BILL\_TO'

AND site\_uses.cust\_acct\_site\_id = acct\_site.cust\_acct\_site\_id

AND NVL(site\_uses.status, 'A') = 'A'

AND cust\_acct.cust\_account\_id = acct\_site.cust\_account\_id

AND acct\_site.cust\_acct\_site\_id = site\_uses.cust\_acct\_site\_id

AND ps.customer\_id = acct\_site.cust\_account\_id

AND ps.customer\_site\_use\_id = site\_uses.site\_use\_id HAVING

NVL(SUM(arr.amount\_applied), 0) > 0;

-- Query for Total Uncleared Receipt Amount

SELECT NVL(SUM(ps.amount\_due\_remaining), 0) total\_uncleared\_receipts

FROM hz\_cust\_accounts\_all cust\_acct,

ar\_payment\_schedules\_all ps,

ar\_receivable\_applications\_all arr,

hz\_cust\_acct\_sites\_all acct\_site,

hz\_party\_sites party\_site,

hz\_locations loc,

hz\_cust\_site\_uses\_all site\_uses,

ar\_cash\_receipt\_history\_all crh,

ar\_cash\_receipts\_all acr,

gl\_code\_combinations cc

WHERE TRUNC(ps.gl\_date) <= :p\_as\_of\_date

AND ps.customer\_id = cust\_acct.cust\_account\_id

AND cust\_acct.account\_number = :p\_account\_number

AND ps.customer\_id = cust\_acct.cust\_account\_id

AND acct\_site.party\_site\_id = party\_site.party\_site\_id

AND loc.location\_id = party\_site.location\_id

AND ps.cash\_receipt\_id = acr.cash\_receipt\_id

AND acr.cash\_receipt\_id = crh.cash\_receipt\_id

AND crh.account\_code\_combination\_id = cc.code\_combination\_id

AND ps.trx\_date <= :p\_as\_of\_date

AND ps.CLASS = 'PMT'

AND ps.cash\_receipt\_id = arr.cash\_receipt\_id

AND arr.status = 'UNAPP'

AND ps.status = 'OP'

AND site\_uses.site\_use\_code = 'BILL\_TO'

AND site\_uses.cust\_acct\_site\_id = acct\_site.cust\_acct\_site\_id

AND NVL(site\_uses.status, 'A') = 'A'

AND cust\_acct.cust\_account\_id = acct\_site.cust\_account\_id

AND acct\_site.cust\_acct\_site\_id = site\_uses.cust\_acct\_site\_id

AND ps.customer\_id = acct\_site.cust\_account\_id

AND ps.customer\_site\_use\_id = site\_uses.site\_use\_id

AND ps.cash\_receipt\_id = crh.cash\_receipt\_id

AND crh.status NOT IN ('CLEARED') HAVING

NVL(SUM(arr.amount\_applied), 0) > 0;

## OPM

### Formula Base Tables and Detail of Products and Ingredients

select a.FORMULA\_ID,a.formula\_no,a.FORMULA\_DESC1,b.INVENTORY\_ITEM\_ID,c.description,b.organization\_id,decode(b.line\_type,-1,’Ingredient’,’Product’) Type

from FM\_FORM\_MST a,FM\_MATL\_DTL b,mtl\_system\_items c

where a.formula\_id=b.FORMULA\_ID

and b.ORGANIZATION\_ID=:your\_Org\_id

and a.FORMULA\_CLASS<>’COSTING’

and b.INVENTORY\_ITEM\_ID=c.inventory\_item\_id

and b.ORGANIZATION\_ID=c.organization\_id

order by a.FORMULA\_ID

Categories: OPM Related Queries

Quantity of Ingredients and Products

June 27, 2010 Faisal Sharif Leave a comment

### Query to Find out Quantity of Products and Ingredients of all Batches in particular Month.

Select b.RECIPE\_DESCRIPTION,a.RECIPE\_VALIDITY\_RULE\_ID,c.INVENTORY\_ITEM\_ID,d.description,decode(c.line\_type,-1,’Ingredient’,’Product’) Type,

sum(e.TRANSACTION\_QUANTITY) quantity

from apps.GME\_BATCH\_HEADER a,apps.gmd\_recipes b,gmd\_recipe\_validity\_rules grr,apps.gme\_material\_details c,apps.mtl\_system\_items d,apps.mtl\_material\_transactions e

where a.FORMULA\_ID=b.FORMULA\_ID

and a.ROUTING\_ID=b.ROUTING\_ID

and a.RECIPE\_VALIDITY\_RULE\_ID=grr.RECIPE\_VALIDITY\_RULE\_ID

and grr.RECIPE\_ID=b.recipe\_id

and a.BATCH\_ID=c.BATCH\_ID

and a.ORGANIZATION\_ID=c.ORGANIZATION\_ID

and c.INVENTORY\_ITEM\_ID=d.INVENTORY\_ITEM\_ID

and c.ORGANIZATION\_ID=d.organization\_id

and a.batch\_id=e.TRANSACTION\_SOURCE\_ID

and a.ORGANIZATION\_ID=e.ORGANIZATION\_ID

and c.INVENTORY\_ITEM\_ID=e.INVENTORY\_ITEM\_ID

and a.batch\_no in (select batch\_no from apps.GME\_BATCH\_HEADER where trunc(plan\_start\_date) between :from\_date and :to\_date)

and a.ORGANIZATION\_ID=:your\_org\_id

and trunc(e.transaction\_date) between :from\_date and :to\_date

group by b.RECIPE\_DESCRIPTION,a.RECIPE\_VALIDITY\_RULE\_ID,c.INVENTORY\_ITEM\_ID,d.description,c.line\_type

order by RECIPE\_DESCRIPTION

## USER RESPONSABILITIES

### Query to Find Responsibilities assigned to particular user.

SELECT b.responsibility\_name NAME

FROM apps.fnd\_user\_resp\_groups a,

apps.fnd\_responsibility\_vl b,

apps.fnd\_user u

WHERE a.user\_id = u.user\_id

AND u.user\_id = (select user\_id from FND\_USER where user\_name=:User\_name)

AND a.responsibility\_id = b.responsibility\_id

AND a.responsibility\_application\_id = b.application\_id

AND SYSDATE BETWEEN a.start\_date AND NVL (a.end\_date, SYSDATE + 1)

AND b.end\_date IS NULL

AND u.end\_date IS NULL

OPM

Formulator for all the Products.

SELECT ffm.formula\_no, ffm.formula\_desc1 formula\_name,

DECODE (fmd.line\_type, 1, 'Product', 'Ingredient') TYPE,

msi.segment1 icode, ood.organization\_code org, fmd.qty,

fmd.detail\_uom,

DECODE (ffm.formula\_status, 700, 'Active', 'New') status

FROM apps.fm\_matl\_dtl fmd,

apps.fm\_form\_mst ffm,

apps.mtl\_system\_items\_b msi,

apps.org\_organization\_definitions ood

WHERE fmd.formula\_id = ffm.formula\_id

AND (fmd.inventory\_item\_id = msi.inventory\_item\_id(+)

AND fmd.organization\_id = msi.organization\_id(+))

AND fmd.organization\_id = ood.organization\_id

ORDER BY formula\_no, DECODE (fmd.line\_type, 1, 'Product', 'Ingredient') DESC;

## OTHER

### Operating Unit name based on the Operating Unit ID

select Business\_group\_id,

Organization\_id,

name,date\_from,date\_to,

legal\_entity\_id,

set\_of\_books\_id

from hr\_operating\_units

where organization\_id = ‘your Operating Unit ID’;