

## Coloane suplimentare

În configurare coloane suplimentare există o serie de tabele legate de lista respectivă. Dacă se dorește legarea altor tabele de cele deja existente, trebuie trecute la TABELA DE JOIN, dat un ALIAS (neaparat nou, să nu mai fie folosit; recomandăm de genul xtextx/xttext/textx). La DESCRIERE CONDIȚII DE JOIN, trebuie completată condiția de legătură dintre tabela nouă și cea existentă deja.

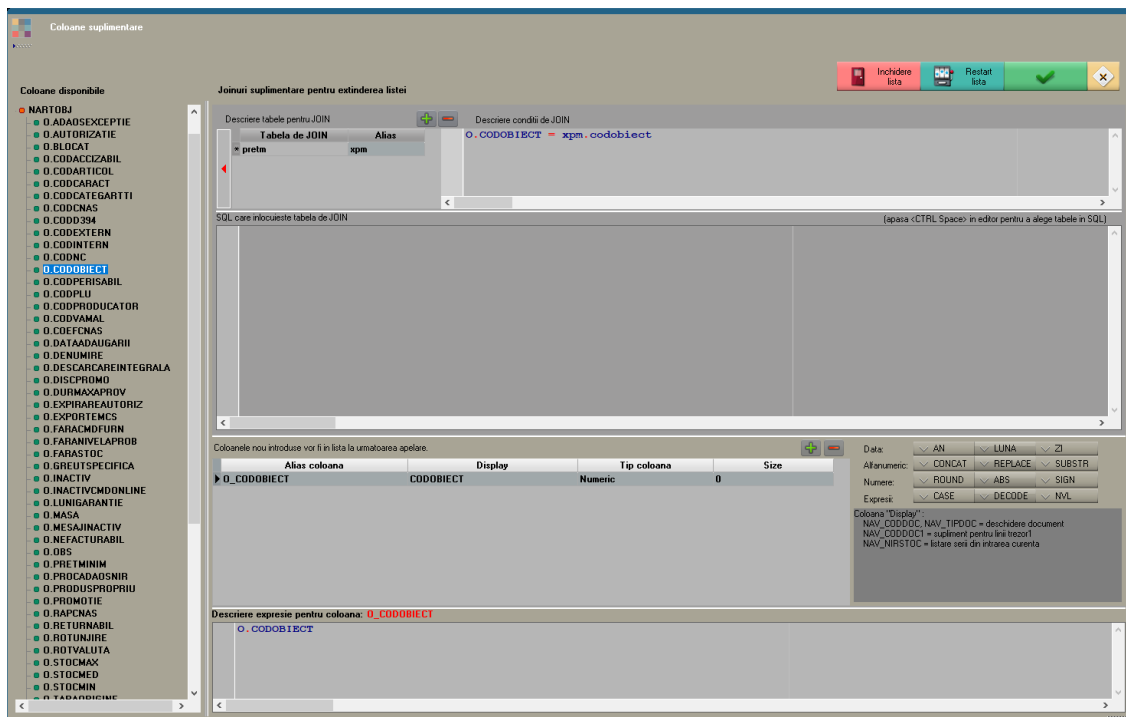


Fig.1

Dacă tabela adusă prin join are mai multe înregistrări pentru o înregistrare din tabela inițială, atunci în listă se vor multiplica liniile existente și coloanele din noua tabelă vor aduce informații diferite. În exemplul de mai sus, dacă un articol are 3 prețuri, în listă va apărea articolul respectiv pe 3 linii și pe coloana Preț vor fi aduse cele 3 prețuri. Dacă vrem să aducă un singur preț, trebuie completată condiția de join să fie legătura unu – la – unu:

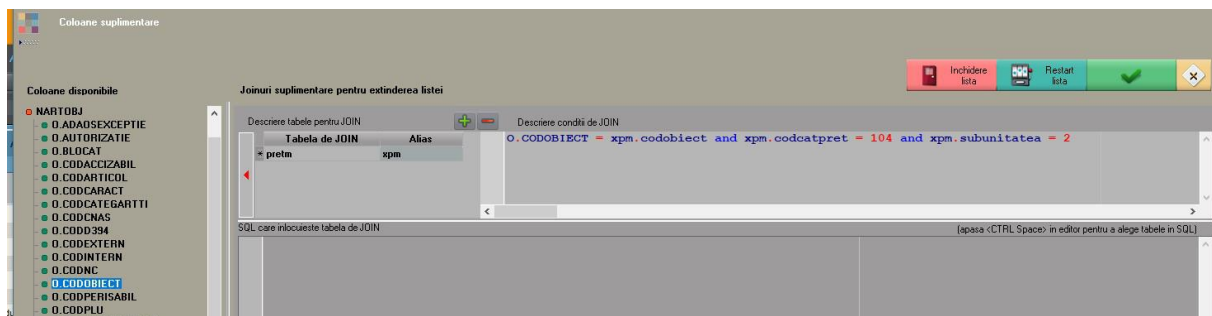


Fig.2

Dacă se dorește introducerea unui SQL de aducere date, trebuie ca acesta să aibă element de join cu unul dintre câmpurile din tabelele deja existente în coloane suplimentare. În exemplul de mai jos s-a adus suma resturilor din NIRSTOC grupate după obiect. Elementul de join este Codobiect.

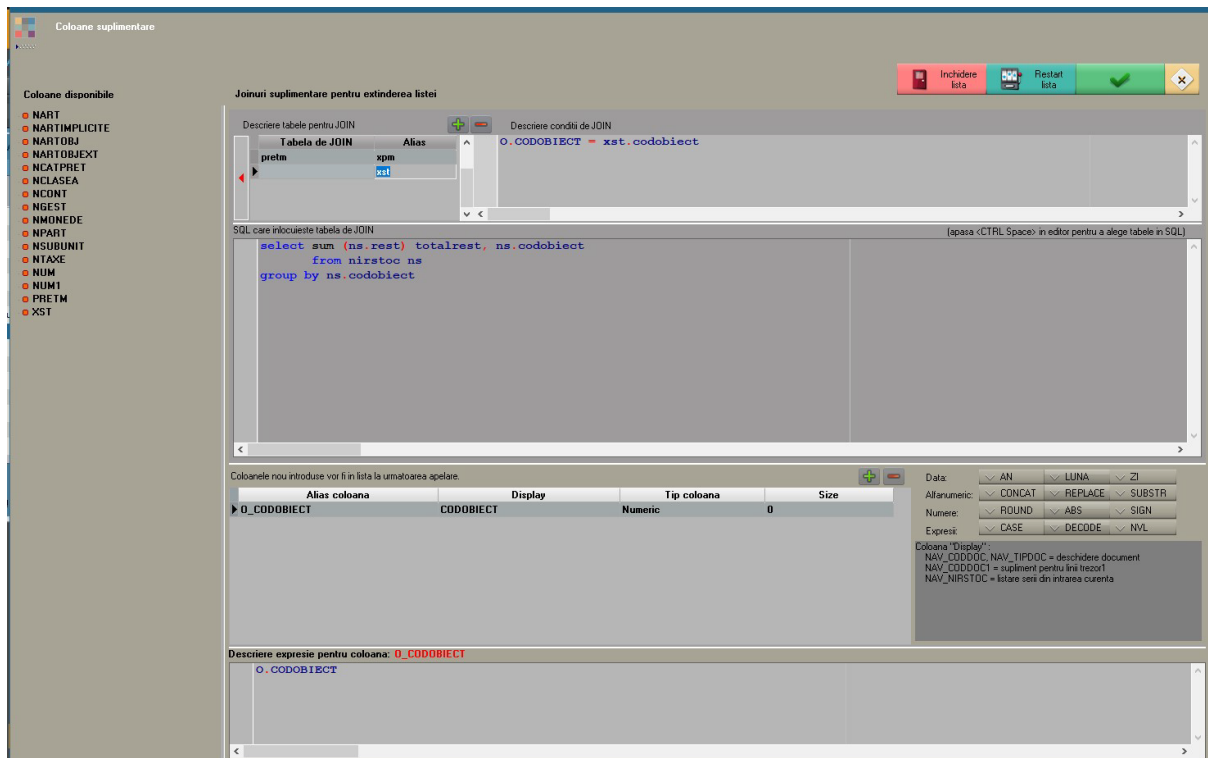


Fig.3

După definirea tabelelor și condițiilor de join se apasă pe săgeata roșie spre stânga pentru a adăuga tabela la **Coloane disponibile**.

Din coloane disponibile se aduc cu dublu-click în **Coloane nou introduse care vor fi în lista la următoarea apelare**.

La **Descriere expresie pentru coloană** se mai pot introduce sql-uri de formatare.

### Exemple:

Concatenare: `AGENT.NUME || ' ' || AGENT.PRENUME`

Concatenare cu trecere la rând nou pentru coloana 2: `col1||chr(13)||col2`

Extragere primele 3 caractere: `SUBSTR(O.CODOBIECT, 0, 3)`

Câmp condiționat: `case when FURN.DEN_PART = TRANSP.DEN_PART then 1 else 0 end`

După definirea coloanelor dorite, trebuie repornită lista pentru a le regăsi la Coloane vizibile > Coloane neafișate.

## Exemple de coloane suplimentare:

- În loc de `CASE WHEN camp= valoare1 camp THEN valoare1 noua WHEN camp= valoare2 camp THEN valoare2 noua ELSE valoare3 noua END`  
se poate folosi `DECODE (camp, valoare1 camp, valoare1 noua, valoare2 camp, valoare2 noua, valoare3 noua)`

```
Decode(SUBSTR(O.CODINTERN, 0, 3)  
, 'ZAG', 'Zagorka SA'  
, 'HKR', 'Heineken'  
, 'URS', 'Ursus'  
, 'HNG', 'Haendler & Natermann'  
, 'CCB', 'Coca-Cola Bulgaria'  
, ''  
)
```

- Coloană suplimentară care aduce prețul de înregistrare mediu din livrări pentru o poziție din ieșiri1 pe listarea facturii de ieșire:

```
(SELECT SUM(xl1.cant * Nvl(xlv.pretinreg, xns.pret))  
FROM livr xl  
INNER JOIN livr1 xl1 on xl1.codlivr = xl.codlivr  
LEFT JOIN nirstoc xns on xns.codnirstoc = xl1.codnirstoc  
LEFT JOIN livrval xlv on xlv.codlivr1 = xl1.codlivr1  
WHERE xl.codies = i.codies  
AND xl.tiplivr = i.tipdoc  
AND xl1.codies1 = i1.codies1) / i1.cant
```

- Coloană suplimentară care aduce datele firmei curente:

alias firmă

condiție 0=0

```
SELECT * FROM TABLE(WINMADMIN.PKG_USERFIRME.FIRMAADR(PKG_GLOBALS.GETCODFIRMA))
```

coloane accesibile orbește pentru moment: denfirma, codfiscal, nrj, persfiz, dentara, simbtara, denjudet, simbjudet, denlocalit, codpostal, strada, nr1, nr2, bloc, scara, etaj, ap, telefon1, telefon2, fax, email, sector, numeadmin, prenumeadmin

- Coloana suplimentară care aduce user-ul documentului:

Tabela de JOIN: `TMP_utilizator`

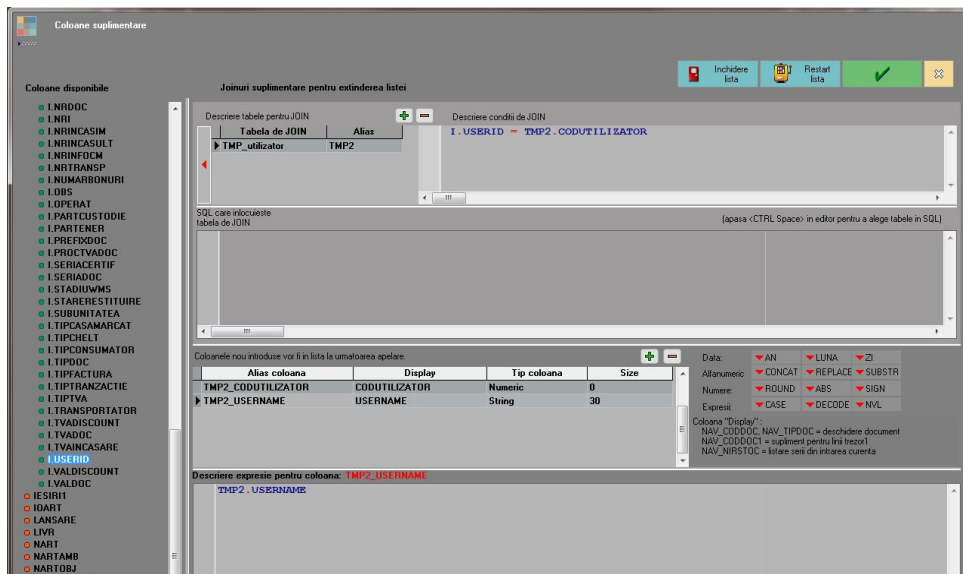


Fig.4

În acest caz, lista nu poate fi folosită în cadrul unui job de tip server pentru că nu sunt date în TMP.

- Concatenare câmpuri:

`AGENT.NUME || ' ' || AGENT.PRENUME`

- Coloana suplimentară pentru a aduce valori din intervale de valabilitate:

`SELECT`

`codioart,`

`Max(codioart1) KEEP (Dense_Rank last ORDER BY ANSTART, LUNASTART) codioart1`

`FROM ioart1`

`WHERE ANSTART * 12 + LUNASTART <= PKG_GLOBALS.GetANLUCRU * 12 +`

`PKG_GLOBALS.GetLUNALUCRU`

`GROUP BY codioart`

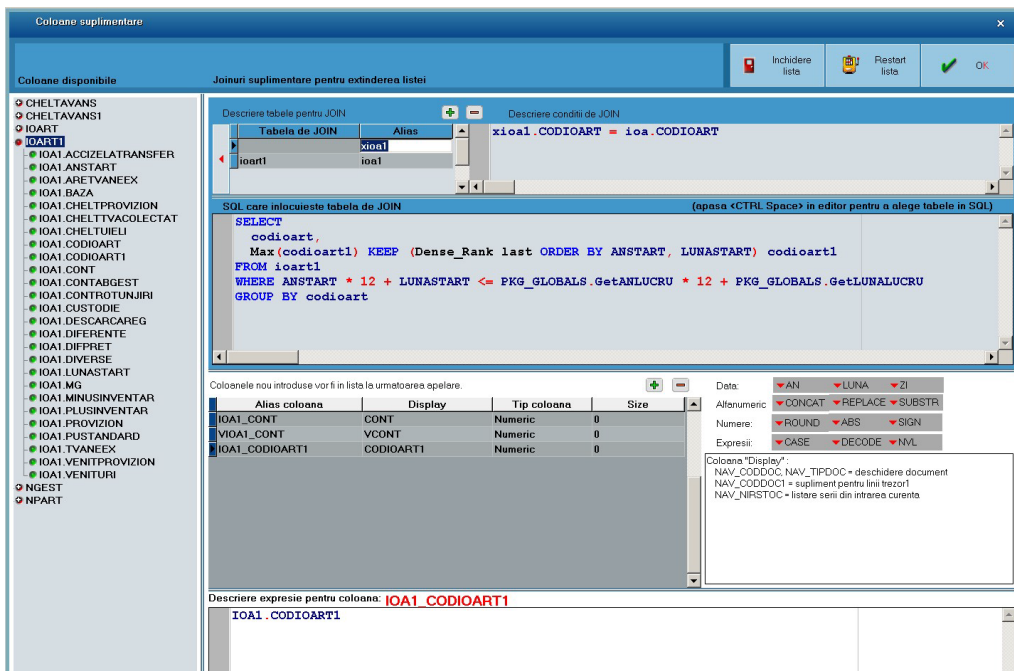


Fig.5

- Coloană care aduce cota de TVA de la articol:

Select codarticol,

max(codarticol1) keep (dense\_rank last order by anstart, lunastart) codarticol1

from nart1

where anstart\*12 + lunastart <= PKG\_Globals.GetAnlucru\*12 + PKG\_Globals.GetLunalucru

group by codarticol

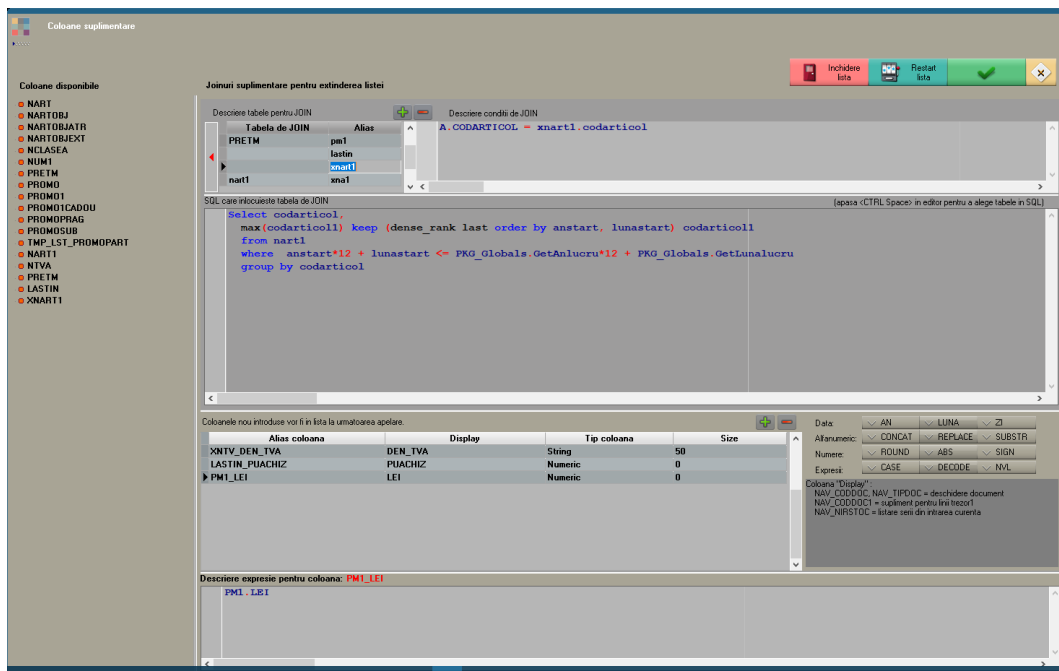


Fig.6

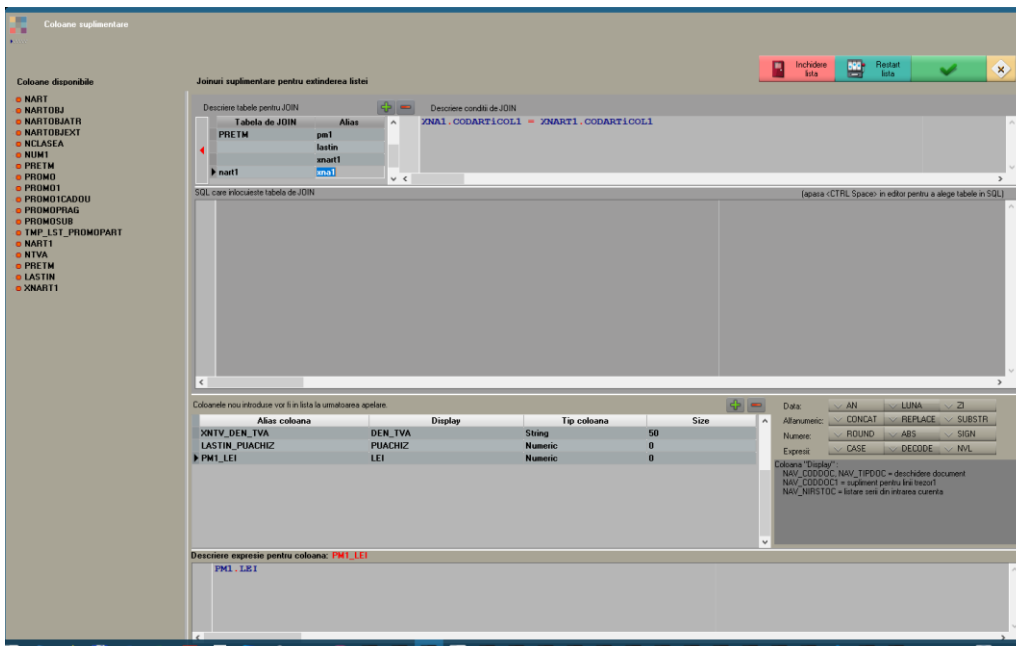


Fig.7

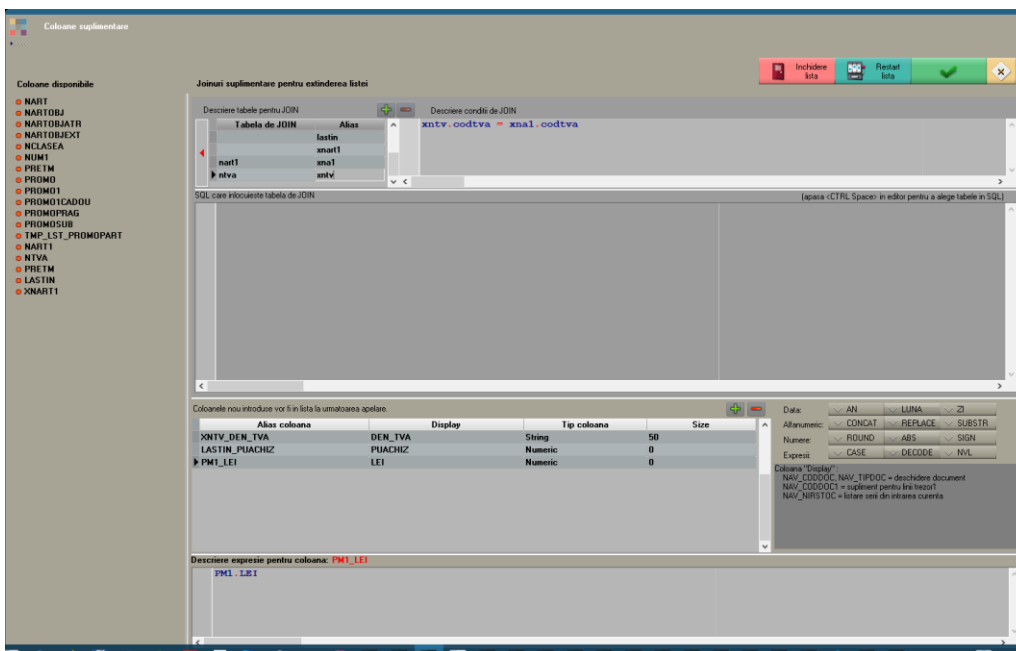


Fig.8

- Coloană care aduce simbolul și denumirea clasei rădăcină aferente clasei unui articol:

SELECT CodClasea, Connect\_By\_Root Simbol SimbRadacina, Connect\_By\_Root den\_clasea  
 denumRadacina

FROM nclasea ca

START WITH Nvl(CodParinte,0) = 0

CONNECT BY codparinte = PRIOR codclasea

SAU:

```
select * from
(
    SELECT connect_by_root codclasea cod, codclasea, nivel, den_clasea, simbol
    FROM nclasea
    CONNECT BY PRIOR codparinte = codclasea
)
```

where nivel = 1

- case when FURN.DEN\_PART = TRANSP.DEN\_PART then 1 else 0 end

- Col suplim : CASE WHEN upper(obs) like '%SINCRONIZ%' THEN 1 WHEN upper(OBS) like '%INCHIDE%' THEN 2 ELSE 0 END

- Coloană suplimentară care aduce pe lista leșiri prețul din oferta aferentă:

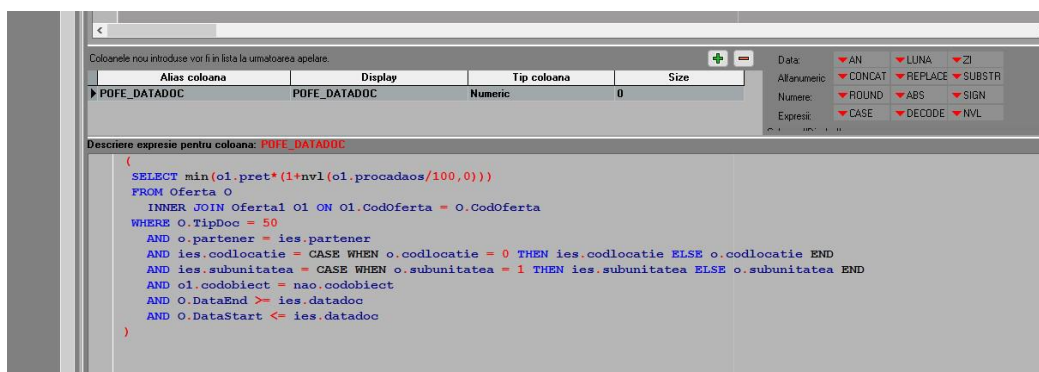


Fig.9

```
(
SELECT min(o1.pret*(1+nvl(o1.procadaos,0)))
FROM Oferta O
    INNER JOIN Oferta1 O1 ON O1.CodOferta = O.CodOferta
WHERE O.TipDoc = 50
    AND o.partener = ies.partener
    AND ies.codlocatie = CASE WHEN o.codlocatie = 0 THEN ies.codlocatie ELSE o.codlocatie END
    AND ies.subunitatea = CASE WHEN o.subunitatea = 1 THEN ies.subunitatea ELSE o.subunitatea END
    AND o1.codobiect = nao.codobiect
    AND O.DataEnd >= ies.datadoc
    AND O.DataStart <= ies.datadoc
)
```

Idem pentru listă Intrări:

```
(
SELECT min(o1.pret*(1+nvl(o1.procadaos/100,0)))
FROM Oferta O
      INNER JOIN Oferta1 O1 ON O1.CodOferta = O.CodOferta
WHERE O.TipDoc = 51
      AND o.partener = i.partener
      AND i.codlocatie = CASE WHEN o.codlocatie = 0 THEN i.codlocatie ELSE o.codlocatie END
      AND i.subunitatea = CASE WHEN o.subunitatea = 1 THEN i.subunitatea ELSE o.subunitatea
      END
      AND o1.codobiect = nao.codobiect
      AND O.DataEnd >= i.datadoc
AND O.DataStart <= i.datadoc
)
```

- Coloană suplimentară care aduce imaginea asociată articolului, la listare etichetă din NIR:

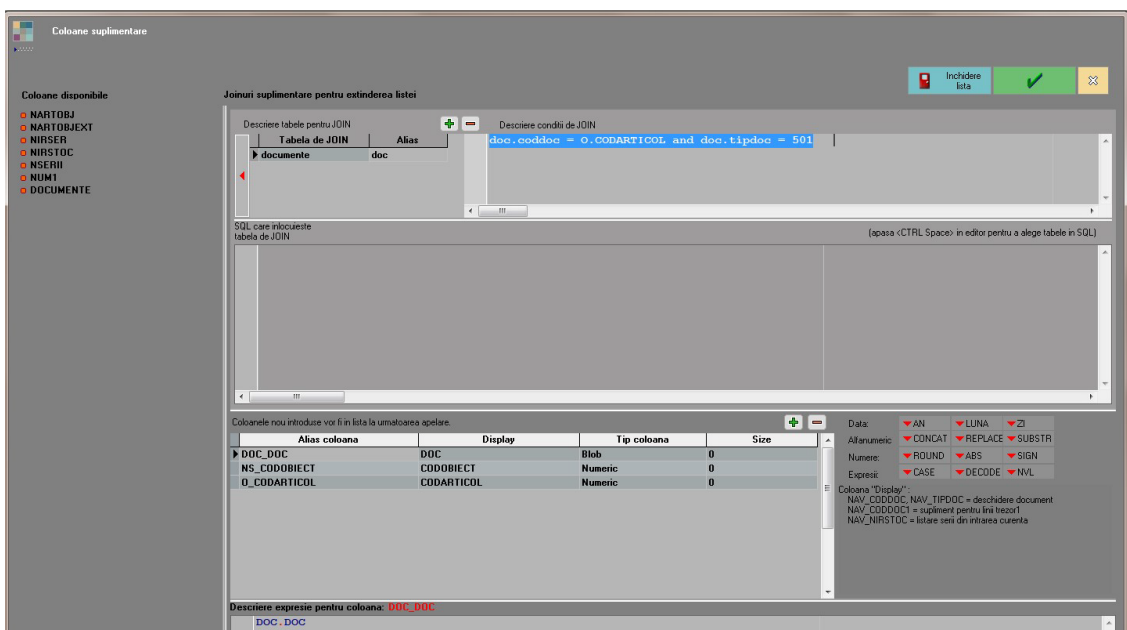


Fig.10



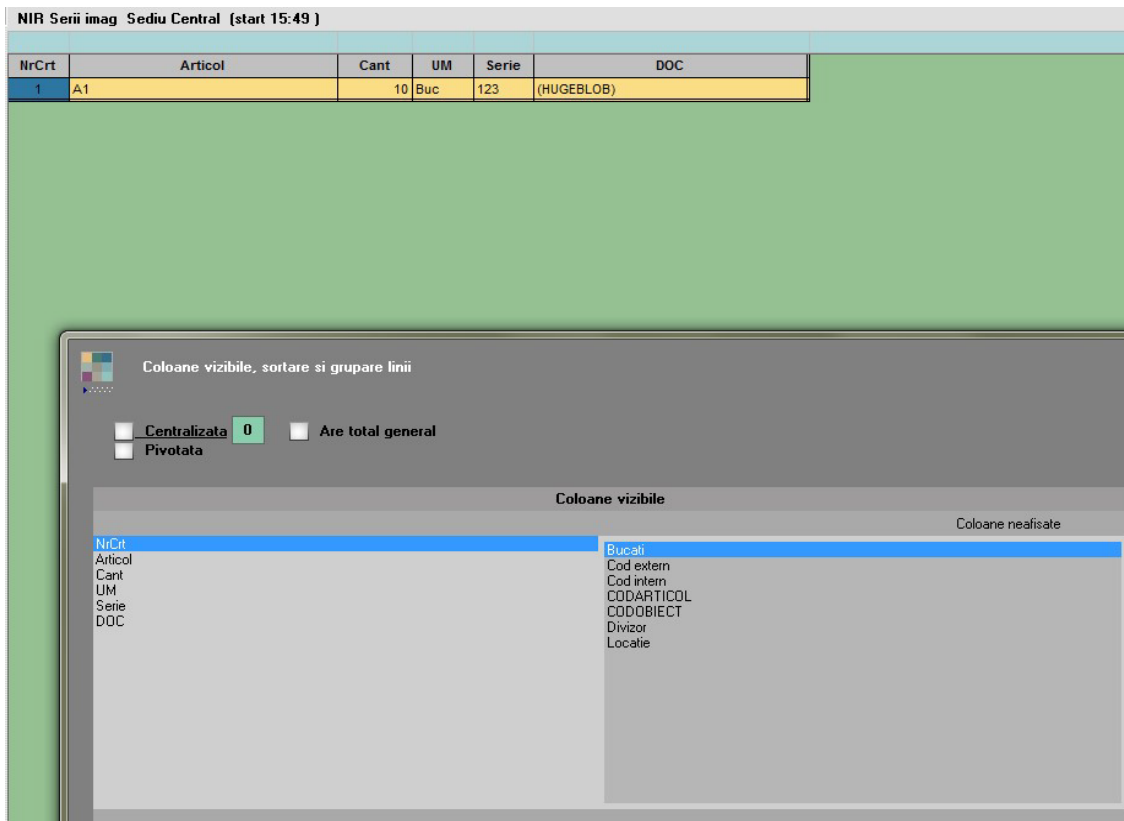


Fig. 11

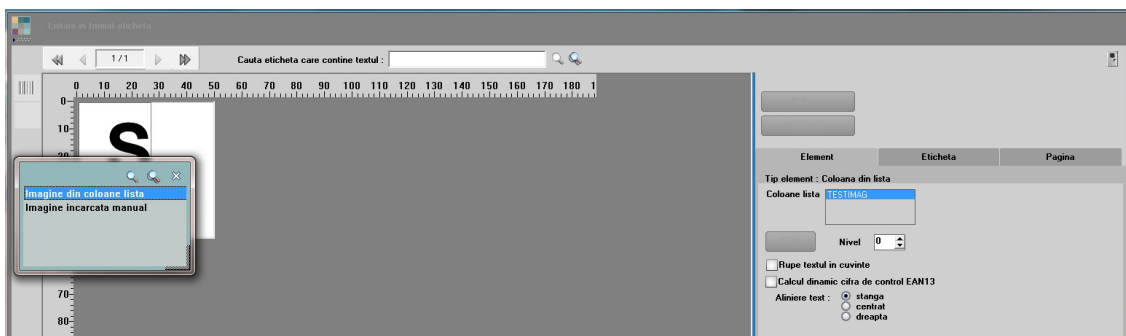


Fig. 12

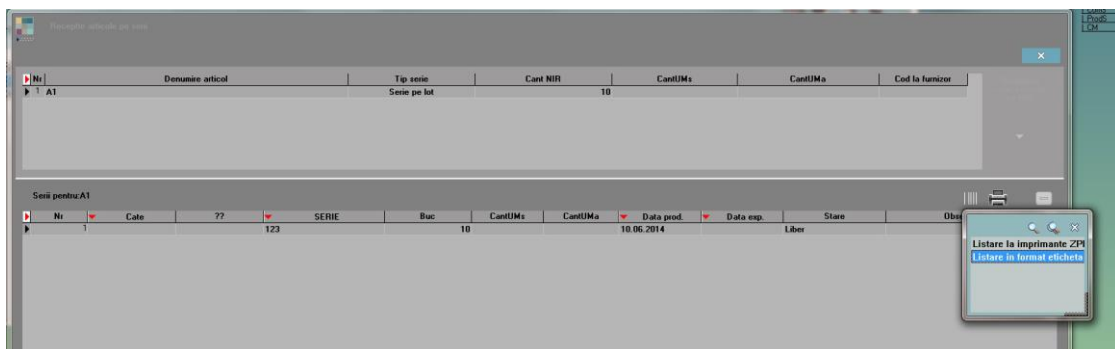


Fig. 13

- Coloană suplimentară care are ultima intrare pentru un obiect: cantitate, preț, dată, partener:

```
select *
  from (select det.codobiect,
              det.cant,
              det.pu,
              intr.datadoc,
              det.codintr1,
              intr.partener,
              row_number() over(partition by det.codobiect order by intr.datadoc desc,
              det.codintr1 desc) as rownumber
        from intrari intr
        join intrari1 det
          on det.codintr = intr.codintr)
 where rownumber = 1
```

- Coloană suplimentară care aduce data ultimei intrări în stoc pe subunitatea și furnizorul de pe Balanța stocurilor pe furnizori (sql pentru descriere coloană suplimentară de tip string de aprox 100 caractere):

```
(
  SELECT distinct LAST_VALUE(ii.nrdoc||'/'||ii.datadoc) OVER (ORDER BY ii.datadoc ROWS
  BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING
FROM intrari ii,
  intrari1 ii1
WHERE ii.partener = furn.codpart and ii1.codobiect = o.codobiect and ii1.codintr = ii.codintr
  and ii.anulat = 0 and ii.operat = 1 and ii.subunitatea = pkg_globals.getsubunitate
```

- Coloană suplimentară care aduce pe listarea de ofertă client o coloană cu gestiunile în care există stoc acoperitor sau parțial:

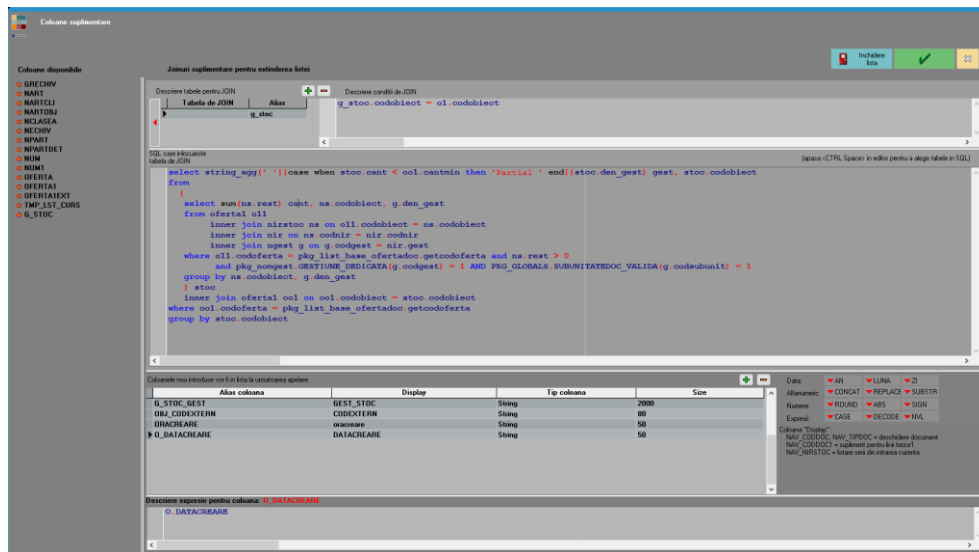


Fig.14

```

select string_agg(' '||case when stoc.cant < oo1.cantmin then 'Partial ' end||stoc.den_gest) gest,
stoc.codobiect
from
(
    select sum(ns.rest) cant, ns.codobiect, g.den_gest
    from oferta1 o11
        inner join nirstoc ns on o11.codobiect = ns.codobiect
        inner join nir on ns.codnir = nir.codnir
        inner join ngest g on g.codgest = nir.gest
    where o11.codoferta = pkg_list_base_ofertadoc.getcodoferta and ns.rest > 0
        and pkg_nomgest.GESTIUNE_DEDICATA(g.codgest) = 1 AND PKG_GLOBALS.SUBUNITATEDOC_VALIDA(g.codsubunit) = 1
        group by ns.codobiect, g.den_gest
    ) stoc
        inner join oferta1 oo1 on oo1.codobiect = stoc.codobiect
    where oo1.codoferta = pkg_list_base_ofertadoc.getcodoferta
    group by stoc.codobiect

```

- Coloană suplimentară care aduce pe listarea de registru jurnal analitic și coloane pentru gestiunea din care se face livrarea ieșirilor:

```

select i.tipdoc, i1.codies1, i1.codies,
    string_agg(distinct coalesce(g.den_gest,gv.den_gest)) gest_livr,
    string_agg(distinct coalesce(g.simbol,gv.simbol)) gests_livr,
    string_agg(distinct coalesce(cg.den_claseg,cgv.den_claseg)) clsg_livr,
    string_agg(distinct coalesce(cg.simbol,cgv.simbol)) clsgs_livr

```

```

from tmp_lst_jurnal1 t
    inner join iesiri i on i.CODIES = t.CODDOC and t.TIPDOC = i.TIPDOC
    inner join iesiri1 i1 on i1.CODIES = i.CODIES and i1.CODIES1 = t.CODDOC1
    inner join livr l on l.CODIES = i.CODIES
    inner join livr1 l1 on l1.CODIES1 = i1.CODIES1 and l1.CODLIVR = l.CODLIVR
    left join nirstoc ns on l1.CODNIRSTOC = ns.CODNIRSTOC
    left join nir on nir.CODNIR = ns.CODNIR
    left join ngest g on nir.GEST = g.CODGEST
    left join nclaseg cg on cg.codclaseg = g.codclasa
    left join livrval lv on lv.CODLIVR1 = l1.CODLIVR1
    left join ngest gv on gv.CODGEST = lv.GEST
    left join nclaseg cgv on cgv.codclaseg = gv.codclasa
group by i.tipdoc, i1.codies1, i1.codies

```

- Coloană suplimentară care aduce restul de pe linia de program de fabricație:  
PF1.CANTPROGRAMAT - nvl((select sum(l.cant) from intrprodlink l where l.codprogramf1 = pf1.CODPROGRAMF1),0)

- Coloană suplimentară care aduce numărul de ordine pe grup în Top Clienți:  
ROW\_NUMBER() OVER (PARTITION BY T.CODCLIENT ORDER BY T.VAL DESC)

sau

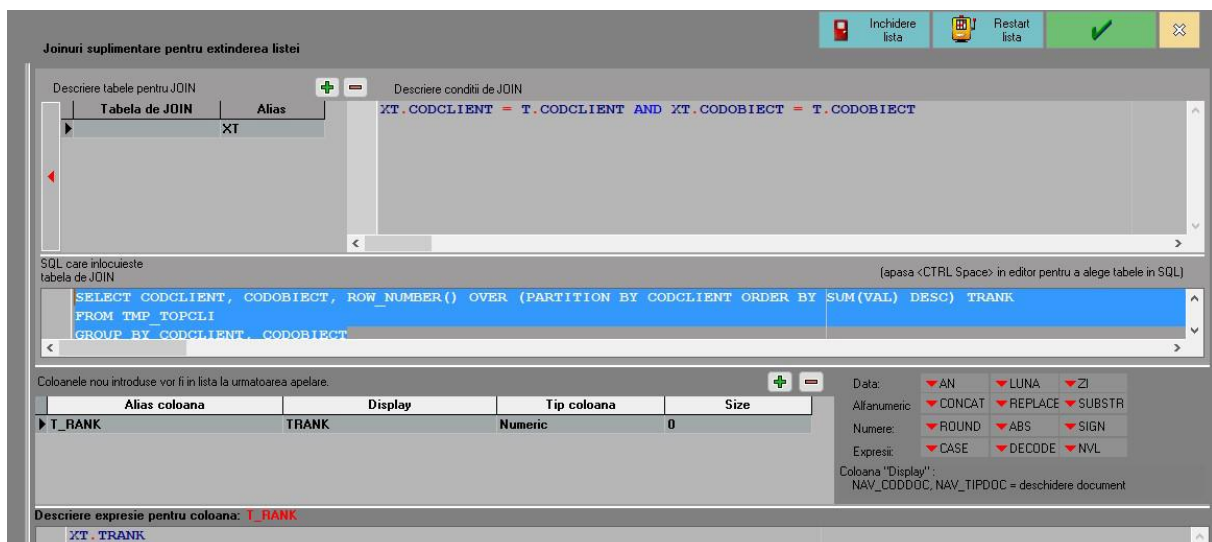


Fig. 15

```

SELECT CODCLIENT, CODOBIECT, ROW_NUMBER() OVER (PARTITION BY CODCLIENT
ORDER BY SUM(VAL) DESC) TRANK
FROM TMP_TOPCLI

```

## GROUP BY CODCLIENT, CODOBIECT

- Coloană suplimentară care aduce stocul la moment mai vechi de 90 zile:

```
select sum (ns.rest) totalrest,  
       ns.codobiect,  
       avg ((trunc(sysdate) - trunc(ns.dataachiz))) zile  
from nirstoc ns  
     inner join nir n on n.codnir = ns.codnir  
     inner join ngest ng on n.gest = ng.codgest  
where ng.codsubunit = 445 and trunc(sysdate) - trunc(ns.dataachiz) >90  
group by ns.codobiect
```

- Coloană suplimentară care aduce stocul la moment:

```
select sum (ns.rest) totalrest, ns.codobiect  
       from nirstoc ns  
     inner join nir n on n.codnir = ns.codnir  
     inner join ngest ng on n.gest = ng.codgest  
where ng.codgest = 457  
group by ns.codobiect
```

- Coloană suplimentară care aduce stocul la moment din subunitatea curentă:

```
select sum (ns.rest) totalrest, ns.codobiect  
       from nirstoc ns  
     inner join nir n on n.codnir = ns.codnir  
     inner join ngest ng on n.gest = ng.codgest  
     inner join nsubunit ns on ns.codsubunit = ng.codsubunit  
where ns.codsubunit = pkg_globals.getsubunitate  
group by ns.codobiect
```

- Coloană suplimentară care aduce ultima intrare:

```
select *  
       from (select det.codobiect,  
                  det.cant,  
                  det.pu,  
                  intr.datadoc,  
                  det.codintr1,  
                  intr.partener,
```

```
row_number () over(partition by det.codobiect order by intr.datadoc desc, det.codintr1
desc) as rownumber
from intrari intr
join intrari1 det
on det.CODINTR = intr.codintr)
where rownumber = 1
```

- Coloană suplimentară care aduce ultimul nirstoc:

```
select *
from (select nsx.codobiect,
nsx.codnirstoc,
nsx.codintr1,
nsx.pret-nsx.adaos puachiz,
row_number () over(partition by nsx.codobiect order by nsx.codnirstoc desc) as rownumber
from nirstoc nsx
)
where rownumber = 1
```

- Coloană suplimentară care aduce ultima modificare a unui document:

```
select *
from (select mx.tablename,
mx.codmaster,
mx.datamodif,
mx.tipmodif,
mx.codmodificare,
row_number () over(partition by mx.codmaster order by mx.codmodificare desc) as
rownumber
from modificari mx
)
where rownumber = 1
```

- Coloană suplimentară care aduce o valoare numerică valabilă pentru tot documentul pe prima linie a acestuia (pe restul liniilor se completeaza 0):

```
CASE WHEN RANK() OVER (PARTITION BY IES.CODIES ORDER BY IES1.CODIES1) = 1 THEN
Coalesce(Ies.NumarBonuri,0) ELSE 0 END
```

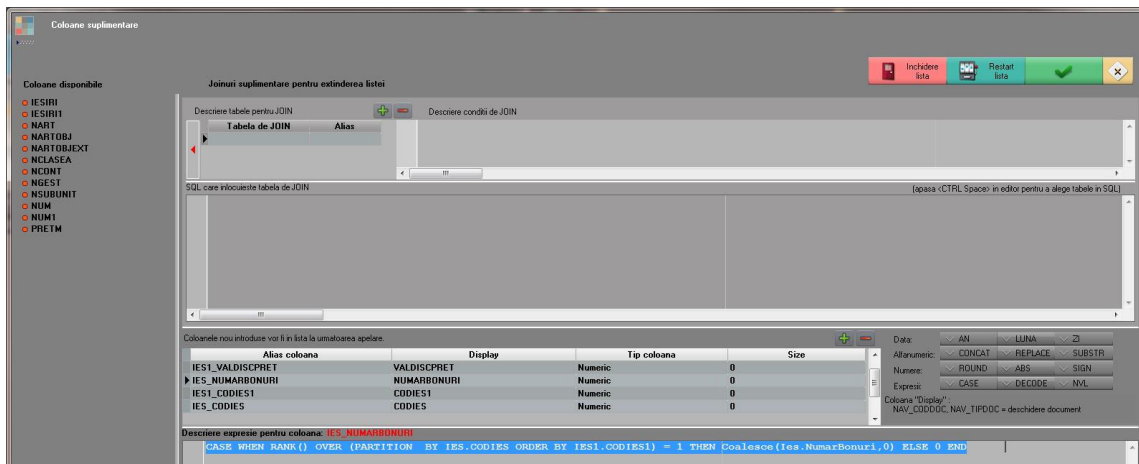


Fig. 16

- Coloană suplimentară pentru cursul cel mai apropiat de data document, la o anumită bancă, într-o anumită monedă:

(select max(valoare) keep (dense\_rank last order by nm1.data)

from nmonede1 nm1

where nm1.CODMONEDA = 104 and nm1.CODBANCA = 111 and nm1.DATA <= i.DATADOC)

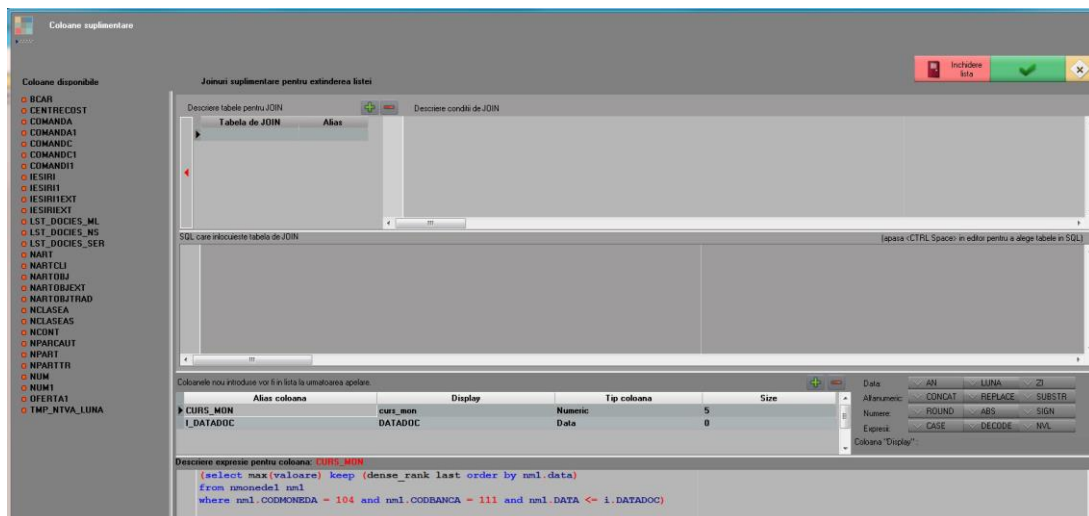


Fig. 17

- Coloană suplimentară care aduce nr crt al liniei pe o coloană numerică nouă (o.denumire = câmp din pachetul listei, nu din alte tabele suplimentare). Lista trebuie să fie ordonată asemenea.

Row\_Number() over (order by o.DENUMIRE)

- Coloană suplimentară care aduce volumul de tranzacții pentru un partener:

PKG\_NOMPART.GETVOLTRANZCLI(<CODPART>)

- Coloană suplimentară care aduce partenerul în fișa de cont (se poate adapta și pentru celelalte liste contabile) :

```

case when FC.OBS like '%-Furnizor :%' then substr(FC.OBS,INSTR(FC.OBS,'Furnizor :') +
10,Length(FC.OBS)-INSTR(FC.OBS,'Furnizor :')-9)
when FC.OBS like '%-Furnizor%' then substr(FC.OBS,INSTR(FC.OBS,'Furnizor') +
8,Length(FC.OBS)-INSTR(FC.OBS,'Furnizor')-7)
when FC.OBS like '%-Client%' then substr(FC.OBS,INSTR(FC.OBS,'Client: ') +
8,Length(FC.OBS)-INSTR(FC.OBS,'Client: ')-7)
when FC.OBS like '%-Retur%' then substr(FC.OBS,INSTR(FC.OBS,'Retur :') +
7,Length(FC.OBS)-INSTR(FC.OBS,'Retur :')-6)
when FC.OBS like '%-Retur Client%' then substr(FC.OBS,INSTR(FC.OBS,'Retur Client :') +
15,Length(FC.OBS)-INSTR(FC.OBS,'Retur Client :')-14)
when FC.OBS like '%-Direct pe venituri%' then 'Direct pe venituri'
when FC.OBS like 'Part: %' then substr((replace(FC.OBS,'Part: ','')),1,instr((replace(FC.OBS,'Part: ','')),',')-1)
when FC.OBS like 'Stingere%' then substr((replace(FC.OBS,'Stingere','')),1,instr((replace(FC.OBS,'Stingere','')),'/')-1)
when FC.OBS like 'Direct pe cont<=>%' then substr((replace(FC.OBS,'Direct pe cont<=>','')),1,instr((replace(FC.OBS,'Direct pe cont<=>','')),'/')-1)
when FC.OBS like 'Client incert%' then substr((replace(FC.OBS,'Client incert','')),1,instr((replace(FC.OBS,'Client incert','')),'/')-1)
when FC.OBS like '%(credit)%' then replace(FC.OBS,'(credit);/','')
when FC.OBS like '%(debit)%' then replace(FC.OBS,'(debit);/','')
when FC.OBS like '%<=>?..?/%' then replace(FC.OBS,'<=>?..?/%','')
when FC.OBS not like '%?..?/%' then fc.obs
else ''
end

```

- Coloană suplimentară care aduce articolele nedescarcate pe intervalul de generare al listei pe Balanța stocurilor pe furnizori:

```

select sum (restiesval) nedescarcate, i1.codobiect
from livr1 l1
inner join iesiri1 i1 on i1.codies1 = l1.codies1
inner join iesiri i on i1.codies = i.codies
where i.subunitatea = pkg_globals.getsubunitate
and i.datadoc between PKG_LIST_SELECTPERIOADA.DataIni and
PKG_LIST_SELECTPERIOADA.DataFin
group by i1.codobiect

```



- Coloana suplimentară care aduce vânzările pe monetare, avize și facturi pe ultimul an pe obiect:

```
Select distinct i1.codobiect, sum(i1.cant) totalvanz
from iesiri i
      inner join iesiri1 i1 on i1.codies = i.codies
where      i.tipdoc      in      (16,19,20)      and      i.datadoc      between
add_months(PKG_LIST_SELFISAMAG.GETDATAREF,      -12)      and
PKG_LIST_SELFISAMAG.GETDATAREF
      and i.operat = 1 and i.anulat = 0
group by i1.codobiect
```

- Liste proprii cu data în interfață:

```
select * from iesiri i where i.tipdoc=19
      and i.An*12+i.LUNA >= PKG_GLOBALS.ANSTART*12+PKG_GLOBALS.LUNASTART
      and i.An*12+i.LUNA <= PKG_GLOBALS.ANSTOP*12+PKG_GLOBALS.LUNASTOP
```

Vezi aici condiția de utilizare filter interfață liste proprii:

```
...CampTipDoc in (Select Valoare from TMP_MANEVRA_LISTE where Filtru = 'DOCS')
...CampCodeGest in (Select Valoare from TMP_MANEVRA_LISTE where Filtru = 'GEST')
...CampCodObiect in (Select Valoare from TMP_MANEVRA_LISTE where Filtru = 'OBJ')
...CampCodPers in (Select Valoare from TMP_MANEVRA_LISTE where Filtru = 'PERS')
...CampCodPAr in (Select Valoare from TMP_MANEVRA_LISTE where Filtru = 'PART')
...CampCodCentru in (Select Valoare from TMP_MANEVRA_LISTE where Filtru = 'CC')
Startlistare:PKG_GLOBALS.ANSTART,PKG_GLOBALS.LUNASTART, PKG_GLOBALS.DataSTART
Stop listare: PKG_GLOBALS.ANSTOP , PKG_GLOBALS.LUNASTOP, PKG_GLOBALS.DataSTOP
Variabila de tip (an*12 + luna) start listare: PKG_GLOBALS.ANLUNASTART
Variabila de tip (an*12 + luna) start listare: PKG_GLOBALS.ANLUNASTOP
```

- Coloană suplimentară care decodifică tip partener din nomenclatorul de parteneri:

```
decode(BITand(n.tippartener,1),1,' intern',") ||
decode(BITand(n.tippartener,2),2,' extern',") ||
decode(BITand(n.tippartener,4),4,' producator',") ||
decode( BITand(n.tippartener,8),8,' asigurator',") ||
decode( BITand(n.tippartener,16),16,' comisionar',")
```

- Coloană suplimentară ce numără facturile pe clienți în Topul clienților:

```
select codclient, count(*) nrfact
from
(select distinct coddoc, tipdoc, codclient
 from tmp_topcli
 )
group by codclient
```

- Coloane suplimentare cu clasele pe niveluri:

```
select nrc.codarticol, nrc.den_articol, nvl(cls.nv1, 'Fara nivel 1') nivel1 , nvl(cls.nv2, 'Fara nivel 2') nivel2
,
nvl(cls.nv3, 'Fara nivel 3') nivel3, nvl(cls.nv4, 'Fara nivel 4') nivel4
from nart nrc
left join (select *
           from (select CONNECT_BY_ROOT codclasea as codclasa, den_clasea, nivel
                 from nclasea
                 connect by codclasea = prior codparinte)
           pivot(max(den_clasea)
                 for nivel in(1 as nv1, 2 as nv2, 3 as nv3, 4 as nv4))) cls
on nrc.clasa = cls.codclasa
```

- Coloane suplimentare cu criteriile de discount pentru a vedea pe fiecare articol ce valoare s-a acordat din respectivul criteriu (în exemplu 394 = codcriteriu). Lista din exemplu este leșiri simplificată. Pentru altă listă trebuie adaptat SQL-ul pentru a se face legătura corectă cu tabelele leșiri și leșiri1

```
pkg_globals.GET_DISC_IES1(coalesce(ies1.codies1,ies1.codies1),coalesce(ies.codies,ae.codies),394)
```

Pe lista leșiri sinteză, trebuie sql-ul de mai jos pentru că datele sunt centralizate pe document:

```
(
select sum(pkg_globals.get_disc_ies1(ix1.codies1, ix1.codies, 394))
from iesiri1 ix1
where ix1.codies = ies.codies
)
```

- Coloane suplimentare cu cota de TVA:

```
PKG_LIST_BASE_UTILS.get_cota_tva_art (codobiect,an,luna)
```

SQL adaptat pe lista Balanță stocuri pe furnizori:

```
PKG_LIST_BASE_UTILS.get_cota_tva_art(o.codobiect,extract(yearfrom  
PKG_LIST_SELECTPERIOADA.DataFin),extract(monthfrom PKG_LIST_SELECTPERIOADA.DataFin))
```

- Coloane suplimentare cu cota de TVA (din joinuri succesive):

xnart1 cu conditia de join: xnart1.codarticol=na.CODARTICOL

```
select
```

```
codarticol,
```

```
Max(codarticol1) KEEP (Dense_Rank last ORDER BY ANSTART, LUNASTART) codarticol1
```

```
FROM nart1
```

```
WHERE ANSTART * 12 + LUNASTART <= PKG_GLOBALS.GetANLUCRU * 12 +  
PKG_GLOBALS.GetLUNALUCRU
```

```
GROUP BY codarticol
```

nart1 cu alias na1 și cu condiția de join: na1.codarticol1= xnart1.codarticol1

ntva cu alias nt și cu condiția de join: nt.codtva=na1.CODTVA

xtva1 cu condiția de join: xtva1.codtva=nt.CODTVA

```
select
```

```
codtva,
```

```
Max(codtva1) KEEP (Dense_Rank last ORDER BY ANSTART, LUNASTART) codtva1
```

```
FROM ntva1
```

```
WHERE ANSTART * 12 + LUNASTART <= PKG_GLOBALS.GetANLUCRU * 12 +  
PKG_GLOBALS.GetLUNALUCRU
```

```
GROUP BY codtva
```

nTVA1 cu alias nt1 și cu conditia de join: nt1.codtva1=xtva1.codtva1

## Review-uri document

Rev. 1.0 12.05.2022 [Creare document](#)

Rev. 1.1 17.11.2023 [Formatare](#)