

- 1) Sur cette exemple de for each avec des each imbriqués:

```
procedure getListeLocataire:
  /*-----
  -----
  Purpose: liste des locataire pour une combinaison type de role
  numero de role
  Notes   : service externe
  -----
  -----*/
  define input parameter pcTypeRole   as character no-undo.
  define input parameter piNumeroRole as int64      no-undo.
  define output parameter table for ttListeLocataire.

  define buffer txrole for txrole.
  define buffer intnt for intnt.
  define buffer tache for tache.

  empty temp-table ttListeLocataire.

  for each intnt no-lock
    where intnt.tpidt = pcTypeRole
      and intnt.noidt = piNumeroRole
      and intnt.tpcon = {&TYPECONTRAT-mandat2Gerance},
      each ctctt no-lock
        where ctctt.tpct1 = intnt.tpcon
          and ctctt.noct1 = intnt.nocon
          and ctctt.tpct2 = {&TYPECONTRAT-Bail},
          each ctrat no-lock
            where ctrat.tpcon = ctctt.tpct2
              and ctrat.nocon = ctctt.noct2
              and (ctrat.dtree = ? or (ctrat.dtree <> ? and ctrat.dtree >
today))):
              create ttListeLocataire.
```

la norminette ne fait pas remarquer que Gilles oublie les buffer ctctt et ctrat.

si maintenant je change le code au niveau du each ctctt (, sur la ligne du each ctctt au lieu de fin de ligne précédente)

```
for each intnt no-lock
  where intnt.tpidt = pcTypeRole
    and intnt.noidt = piNumeroRole
    and intnt.tpcon = {&TYPECONTRAT-mandat2Gerance}
  , each ctctt no-lock
    where ctctt.tpct1 = intnt.tpcon
      and ctctt.noct1 = intnt.nocon
      and ctctt.tpct2 = {&TYPECONTRAT-Bail},
```

la norminette fait bien remarquer à Gilles qu'il oublie le buffer ctctt (et que la virgule serait mieux ailleurs).

2) Ce bout de code :

```
    for each ttMutation
      where ttMutation.CRUD = "U"
        and ttMutation.iNumeroMandatAcheteur = 0
      , each ttListeLotMutation
        where ttListeLotMutation.iNumeroContratMutation =
ttMutation.iNumeroContratMutation
          and ttListeLotMutation.iNumeroBail <> 0
      , last tache no-lock
        where tache.tpcon = {&TYPECONTRAT-bail}
          and tache.nocon = ttListeLotMutation.iNumeroBail
          and tache.tptac = {&TYPETACHE-quittancement}
          and tache.dtfm = ?
      , first ctrat no-lock
        where ctrat.tpcon = tache.tpcon
          and ctrat.nocon = tache.nocon
  // todo est-ce bien nécessaire de les trier ??
    by ttListeLotMutation.iNumeroContratMutation
    by ttListeLotMutation.iNumeroBail descending:
      mLogger.writeLog (0, substitute("Validation - Mutation &1 Lot &2 bail actif &3 -
&4", ttMutation.iNumeroContratMutation, ttListeLotMutation.iNumeroLot,
ttListeLotMutation.iNumeroBail, ctrat.lbnom)).
      mError:createError({&information}, 1000683, substitute('&2&1&3', separ[1],
ttListeLotMutation.iNumeroBail, ctrat.lbnom)). //Il existe des baux actifs sans nouveau
mandat de gérance (&1 - &2). Ces baux seront résiliés et la reprise mandat, UL et baux devra
se faire manuellement
      leave boucleRechercheSiLotActif.
    end.
```

donne 2 alertes de type :

Bien entendu, cela ne gêne pas l'utilisation de l'outil.

Gilles.

