

Desvendando o Select Dinâmico

20/12/2010 11:47

Qual programador nunca se deparou com aquela regra funcional complicada (ou chata), que faz você ter que selecionar um monte de **tabelas quase iguais**, todas com o mesmo campo e com cláusula where parecidas?

Antes você sofria, mas depois desse post você **não deverá sofrer mais!** (claro, sempre tem aquele nego chato que curte um CTRL+C CTRL+V, mas beleza, cada um na sua!).

Report com exemplo de como montar **selects dinâmicos**:

```
REPORT zselect_dinamico.

* Declarações
*-----
DATA: fieldcat      TYPE lvc_t_fcat,
      r_fieldcat    LIKE LINE OF fieldcat,
      d_reference   TYPE REF TO data.

FIELD-SYMBOLS: <table> TYPE STANDARD TABLE,
               <wa_table> TYPE ANY.

TYPES: BEGIN OF ty_select,
        line TYPE fieldname,
      END OF ty_select.

DATA: t_campos TYPE TABLE OF ty_select WITH HEADER LINE,
      t_tables TYPE TABLE OF ty_select WITH HEADER LINE,
      t_where  TYPE TABLE OF ty_select WITH HEADER LINE.

* Criação do Fieldcat para Tabela Dinâmica
*-----
r_fieldcat-fieldname = 'KUNNR'.
r_fieldcat-ref_field = 'KUNNR'.
r_fieldcat-ref_table = 'KNA1'.
APPEND r_fieldcat TO fieldcat.

r_fieldcat-fieldname = 'NAME1'.
r_fieldcat-ref_field = 'NAME1'.
r_fieldcat-ref_table = 'KNA1'.
APPEND r_fieldcat TO fieldcat.

* Método Estático que cria a tabela interna do tipo do fieldcat
* Esse método está sendo "emprestado" da classe de ALV, só para
* a montagem da tabela interna de maneira dinâmica
*-----
CALL METHOD cl_alv_table_create=>create_dynamic_table
  EXPORTING
    it_fieldcatalog = fieldcat
  IMPORTING
    ep_table       = d_reference.

* A variável d_reference é do tipo DATA
* Associa a referência da tabela, para ser possível alterar os dados.
*-----
ASSIGN d_reference->* TO <table>.

* Monta tabela com os Campos que serão selecionados
```

```

*-----
LOOP AT fieldcat INTO r_fieldcat .
  t_campos-line = r_fieldcat-fieldname.
  APPEND t_campos.
ENDLOOP.

* Tabela onde será feito o Select
*-----
t_tables-line = 'KNA1'.

* Cláusula Where do Select
*-----
t_where-line = 'KUNNR = ''0000006251'' OR'.
APPEND t_where.
t_where-line = 'LAND1 = ''CL'' '.
APPEND t_where.

* Select Totalmente Dinâmico! :)
*-----
SELECT (t_campos)
  INTO TABLE <table>
  FROM (t_tables-line)
  WHERE (t_where).

* Divirta-se vendo os dados nos field-symbols
*-----
LOOP AT <table> ASSIGNING <wa_table>.

  BREAK-POINT.

ENDLOOP.

BREAK-POINT.

```

Um ponto importante é lembrar que a montagem de selects dinâmicos, em alguns poucos casos, prejudica a performance do programa (normalmente em selects *monstros*). Eu não acreditava muito nisso até fazer testes massivos de comparação de códigos e encontrar alguns problemas. Mas fique tranquilo, porque normalmente coisas dinâmicas quebram o seu galho e não atrapalham em nada!

Utilize o código acima pra **consulta** ou mesmo **copie e cole num programa local** para ver como funciona. Se você quiser, pode trocar as tabelas, os campos, a cláusula where... 😊

Abraços!

Comentários

Rafael Castro Melo — 17/07/2014 11:28

Bom dia,

Muito show o abap zombie... preciso de ajuda... tenho experiência em desenvolvimento em advpl-Totvs e estou iniciando no abap, porém estou com uma dúvida...

preciso somar o valor de duas colunas no select
 select (data_emissao + cond_pgto) as data_vencimento
 from tabela

como eu posso fazer isso?

Atenciosamente

Rafael Castro Melo

Mauricio Cruz — 22/07/2014 09:38

Não dá direto no SELECT Rafael. Você vai ter que trazer para o programa, e somar as colunas direto na tabela interna.

ABs!

Paulo Sales — 05/12/2012 13:28

Muito bom!

Parabéns pelo post.

Este site para mim é o melhor aqui do Brasa...

Mauricio Cruz — 05/12/2012 14:24

Obrigado Paulo, que bom que você curte nosso trabalho 😊

Abraços!

crisovao sousa — 29/10/2012 13:38

Olá ..

TEm como colocar a clausula " for all entries in " dinamicamente no select ? estou tentando implementar o código abaixo.. mais não ta dando certo, tem alguma dica ?

```
SELECT (g_selfields_tab)
INTO TABLE
FROM (pv_tabla)
FOR ALL ENTRIES IN (tg_filtro)
WHERE (gt_where).
```

Mauricio Cruz — 29/10/2012 13:47

Essa eu acho que eu sei 😊

Tente colocar o FOR ALL ENTRIES t_tabela na variável que você colocar dinamicamente no FROM.

No seu exemplo, a string "PV_TABELA" teria que ter não só a tabela do from, mas também o for all entries.

```
PV_TABELA = 'MARA FOR ALL ENTRIES IN T_FAE'.
```

E daí você complementa as variáveis do where c/ as igualdades para T_FAE, como num for all entries normal.

Acho que é isso, tente aí e depois nos conte se deu certo. Abs!

crystovao sousa — 29/10/2012 14:06

Consegui fazer assim :

* executa o select com a clausula FOR ALL ENTRIES.

```
SELECT (g_selfields_tab)
INTO TABLE
FROM (pv_tabla)
FOR ALL ENTRIES IN
WHERE (gt_where).
```

A minha dificuldade agora .. é mover a tabela interna que contem as linhas que quero usar no for all entries in, para o FIELD-SYMBOLS

Como movo esses valores ?

Mauricio Cruz — 30/10/2012 07:18

Só pra constar, se você escreveu o FOR ALL ENTRIES no select, ele deixa de ser completamente dinâmico. Se hora o programa tiver que executar sem for all entries, e hora tiver que executar com o for all entries, você vai ter que escrever duas vezes o select.

E o pior: para você conseguir fazer esse FOR ALL ENTRIES que você quer, c/ o Field-Symbol, obrigatoriamente seu Field-Symbol vai ter que referenciar o tipo da tabela do FOR ALL ENTRIES (a que você daria o ASSIGN). Se você não fizer isso, não vai compilar, porque o WHERE pede que você especifique o campo do FOR ALL ENTRIES (ex.: WHERE mara IN t_fae-mara).

Dependendo do seu requerimento, você pode pesquisar sobre GENERATE SUBROUTINE POOL. Não dá pra explicar isso aqui, mas você consegue vários exemplos na internet.

Abs!

Mauricio Cruz — 28/11/2012 10:37

Update: fiz um post sobre isso 😊

<http://www.abapzombie.com/dicas-abap/2012/11/28/gerando-e-executando-codigos-em-tempo-de-execucao/>

Mauro Laranjeira — 28/09/2011 14:09

Certo Mauricio,

Pra ajudar um brother fiz um esquema bem parecido de select dinâmico, mas sem a clausula where.

Creio que só ira acrescentar, segue abaixo:

```
FIELD-SYMBOLS: <fs_tabela> TYPE STANDARD TABLE.
```

```
DATA: tl_tabela TYPE REF TO data.
```

```
DATA: vl_tabela TYPE string VALUE 'MARA'.

CREATE DATA tl_tabela TYPE TABLE OF (vl_tabela).

ASSIGN tl_tabela->* TO <fs_tabela>.

SELECT * UP TO 10 ROWS
  FROM (vl_tabela)
  INTO TABLE <fs_tabela>.

BREAK-POINT.
```

Abs a todos =D
Mauro Laranjeira

Mauricio Cruz — 28/09/2011 14:12

Seu exemplo é legal porque, se a tabela interna tiver a mesma definição de uma tabela standard, você consegue fazer direto com o CREATE DATA.

É nós mano Mauro!

Renan William Alves — 11/08/2011 18:43

Opa! O blog está ótimo!

Sempre procuro material para ABAP em sites brazucas e é muito difícil encontrar algo de qualidade, vocês e o ABAP101 estão de parabéns 😊

Só estou sentindo falta de uns botões aqui no final do post para recomendar para os amigos! Vai lá no painel do WP e busca um plugin de share pra resolver isso 😊

Abraços

Mauricio Cruz — 01/09/2011 11:50

Apanhei um pouco porque não sei nada de PHP, mas consegui, finalmente, arrumar o ponto que o Renan comentou 😊

Abraços e valeu pela sugestão!