

ABAPZombie Guide to ABAP – Parte 26 – CLEAR

25/07/2011 10:00

Ah, o CLEAR. Que programador nunca ficou debuggando 5 horas para descobrir que “faltava um CLEAR”? Eu já 😞

Tristezas dignas do [@ABAPdepressao](#) à parte, vamos dar uma olhada nesse comando simples e comum no nosso dia-a-dia.

O CLEAR limpa. Preciso dizer mais alguma coisa?

```
REPORT    zombie_clear.

DATA: v_var TYPE i VALUE 10.

DATA: wa_struc TYPE mara.

DATA: t_table TYPE TABLE OF mara.

DATA: t_table_h TYPE TABLE OF mara WITH HEADER LINE.

* Limpou a variável
CLEAR v_var.

* Limpou a estrutura
CLEAR wa_struc.

* Limpou a tabela
APPEND INITIAL LINE TO t_table.
CLEAR t_table.

* Limpou a tab... O QUE! COMO ASSIM?!? NÃO LIMPOU?
APPEND INITIAL LINE TO t_table_h.
CLEAR t_table_h.
```

Pois é, na verdade eu preciso dizer algumas coisas sim 😞

O **CLEAR** vai levar a variável à seu valor inicial, de acordo com as regras do ABAP. O valor colocado com a adição VALUE para definir um valor inicial na declaração, é ignorado. (Ex. numa variável CHAR de 1 posição e valor default A, tem como valor inicial um espaço, ou o valor hexadecimal 20).

O CLEAR limpa várias coisas, mas tome cuidado para não confundir o funcionamento nas **tabelas com e sem header line**.

Em tabelas **com header line**, o CLEAR irá limpar a linha de header da tabela, e não o seu conteúdo. Caso a tabela **não tenha header line**, como é o caso de T_TABLE no código acima, o CLEAR irá limpar o conteúdo da tabela. Rode o programa para testar 😊

Você ainda pode utilizar a adição **WITH** para “limpar” a variável utilizando o valor de outra variável, ou a adição **WITH NULL** para apagar de vez os valores hexadecimais da variável (vai ficar tudo 00).

E ainda tem mais: o **CLEAR faz quase a mesma coisa que o comando FREE**, ou seja, ele elimina o valor da variável da memória do programa. Porém, os comandos diferem um pouco no tratamento de tabelas internas, onde o **CLEAR vai limpar a tabela mas não vai liberar a memória ocupada pelas linhas da tabela**. Ou seja, você vai limpar o conteúdo, mas a área de memória vai continuar sendo ocupada pelo seu programa.

E quem diria que um comando tão simples teria tantas coisas para explicar, não é? 😊

Abraços!