

## ABAPZombie Guide to ABAP – Parte 22 – CALL METHOD

06/04/2011 09:00

O CALL METHOD chama um método de uma classe. Dãr.

Mas, caro amigo que não conhece OO, esse simples comando pode tornar-se caótico caso você fique perdido nos conceitos do **ABAP Objects**.

Portanto, vou passar por diversas formas de chamar um método com o CALL METHOD, inclusive chamadas dinâmicas.

Lembrando que eu, particularmente, não utilizo a sintaxe do CALL METHOD pra quase nada, já que você pode omitir a palavra reservada e chamar o método de uma outra forma. Em todas as explicações, vou colocar as duas formas, e você usa a que der na telha, beleza? 😊

Vamos lá!

1 – Chamada de método de Instância, sem assinatura:

```
*-----*
* 1 - Chamada de método de Instância, sem assinatura:

* Chamada com o CALL METHOD
CALL METHOD o_zombie->criar_zumbi_que_anda.

* Omitindo o CALL METHOD
o_zombie->criar_zumbi_que_anda( ).
```

2 – Chamada de método Estático, sem assinatura

```
*-----*
* 2 - Chamada de método Estático, sem assinatura

* Note o sinal de = para chamar itens estáticos
* Na chamada estática, não existe referência da instância (O_ZOMBIE)
* mas sim da classe local LCL_ZOMBIE

* Chamada com o CALL METHOD
CALL METHOD lcl_zombie=>criar_zumbi_estatua.

* Omitindo o CALL METHOD
lcl_zombie=>criar_zumbi_estatua( ).
```

3 – Chamada de método dinâmico, sem assinatura:

```
*-----*
* 3 - Chamada de método dinâmico, sem assinatura:

* Sim, dá pra chamar métodos de forma dinâmica
* Só não dá para omitir o CALL METHOD nas chamadas dinâmicas.

DATA: l_metodo TYPE char30.

l_metodo = 'CRIAR_ZUMBI_ESTATUA'.

CALL METHOD o_zombie->(l_metodo).
```

```
l_metodo = 'CRIAR_ZUMBI_QUE_ANDA'.
```

```
CALL METHOD o_zombie->(l_metodo).
```

*\* E os parâmetros? É só escrever, o compilador ignora a sintaxe  
\* dos parâmetro de métodos dinâmicos. Escreva direito ou DUMP! :)*

#### 4 – Chamada de método de Instância, com parâmetro de importação e exportação:

```
*-----*
* 4 - Chamada de método de Instância, com parâmetro de
* importação e exportação:

DATA: l_comeu TYPE char1.

* Sem omitir o CALL METHOD, fica bem parecido com o CALL FUNCTION
CALL METHOD o_zombie->comer_cerebros
  EXPORTING
    p_quantidade = '4'
  IMPORTING
    p_comeu       = l_comeu.

* Omitindo, vc escreve os parâmetros dentro dos ( )
o_zombie->comer_cerebros( EXPORTING
                          p_quantidade = '0'
                          IMPORTING
                          p_comeu      = l_comeu ).
```

#### 5 – Chamada de método de Instância, com parâmetro retorno:

```
*-----*
* 5 - Chamada de método de Instância, com parâmetro retorno:

DATA l_balas TYPE char30.

* Returning parece o EXPORTING, mas método só podem retornar UM
* parâmetro, e nenhum mais. Lembrando que retornar <> exportar
* para métodos, hehehehehehe

* Tô com medo, me dá munição pra matar zumbi
CALL METHOD o_zombie->me_da_municao
  RECEIVING
    p_balas = l_balas.

WRITE: / l_balas.

* E o jeito mágico de receber direto na variável.
* Na chamada do método abaixo, vale o que eu expliquei nos tópicos
* anteriores
l_balas = o_zombie->me_da_municao( ).

WRITE: / l_balas.

* Você pode receber o valores de um parâmetro RETURNING direto numa
* variável do mesmo tipo do parâmetro. (H) -> CTRL+C CTRL+V no MSN
```

#### 6 – Chamada de método de Instância, com exceções:

```
*-----*
* 6 - Chamada de método de Instância, com exceções:

CALL METHOD o_zombie->me_da_uma_bazuca
  EXCEPTIONS
```

```

    nao_tem = 1.

    IF sy-subrc <> 0.
        WRITE: / 'Se vira com a DOZE, campeão!'.
    ENDIF.

    o_zombie->me_da_uma_bazuca( EXCEPTIONS nao_tem = 1 ).

    IF sy-subrc <> 0.
        WRITE: / 'Se vira com a DOZE, campeão!'.
    ENDIF.

```

Acredito que esses são os modos mais usados. Para as chamadas com parâmetros, tudo que está escrito na explicação dos métodos de instância, vale para os métodos estáticos e na chamada dinâmica.

Não vou falar de Classes de Exceções aqui, isso provavelmente vai estar no post do comando CATCH.

E se você quiser executar os comandos acima, não precisa ficar dando CTRL+C CTRL+V não camarada: [clique aqui para baixar o TXT com todo o código](#), além de poder ver meus comentários infâmes e malucos 😊

Abraços!

## Comentários

**Thiago Freu** — 04/04/2013 14:54

HAHAHAHAHAHAHAHAHAHA Simplesmente o melhor:

"WRITE: / 'Se vira com a DOZE, campeão!'"

Post show de bola d+! Muito simples de pegar o conceito e com um humor ninja! Parabéns mesmo!!!  
=D

**L.Chada** — 12/07/2012 14:50

Ainda não estava completamente convencido a entrar de cabeça no OO...

É até engraçado, mas as vezes a gente precisa de só mais um detalhe, um pequeno argumento para se render!

\* E o jeito mágico de receber direto na variável.

l\_balas = o\_zombie->me\_da\_municao( ).

De agora pra frente, compromisso assumido, novos desenvolvimentos só em OO!

**Vinicius Ostan** — 16/12/2011 14:03

Como sempre, nos ajudando forma clara e divertida.... Muito obrigado!!!