

Sumario:

Este tutorial é descreve a criação básica de um REPORT exemplificado com uma REPORT que lógica de calculadora básica Orientada a Objetos.

Sobre o Autor:

Uderson Luis Fermino, formado em Ciências da Computação pela Faculdade de Pesquisa e Ensino IPEP, atua no mercado a 2 anos como desenvolvedor Java nas plataformas: (J2SE, J2EE e J2ME), com participação em grandes projetos envolvendo estas tecnologias. É consultor ABAP com experiências em REPORT, ALV (GRID, LIST, BLOCK, OO, TREE, HIERARQUICK), IDOC, ALE, ONLINE, SAPSCRIPT, SMARTFORM, NETWEAVER (JCO, BSP, WebDynpro).

Email:

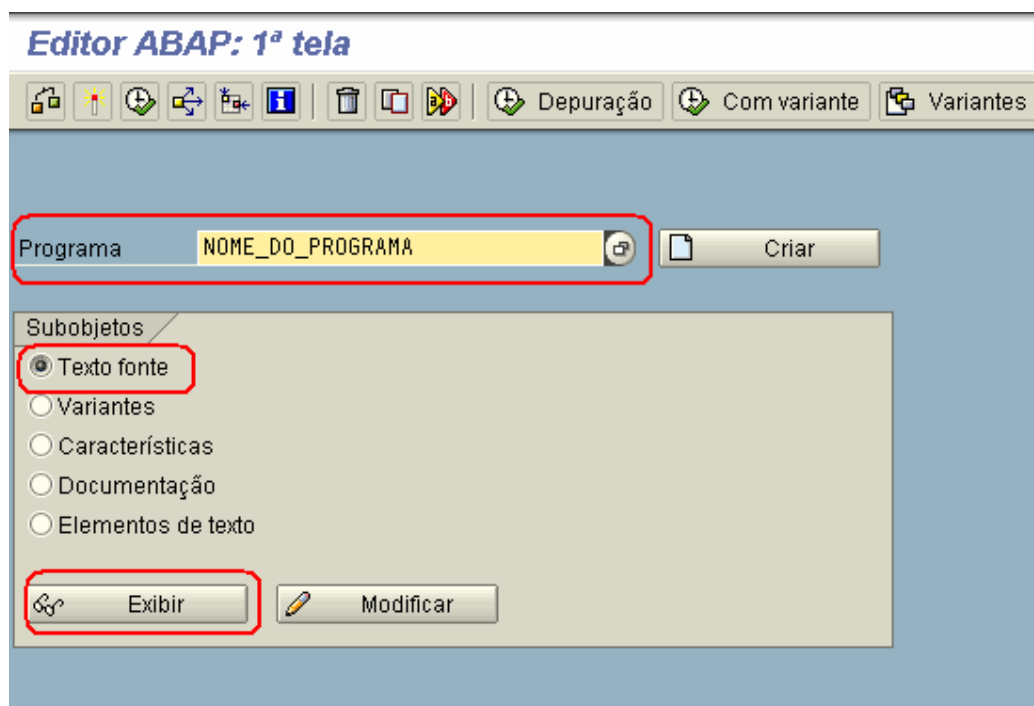
uderson@gmail.com

Entre na transação SE38:



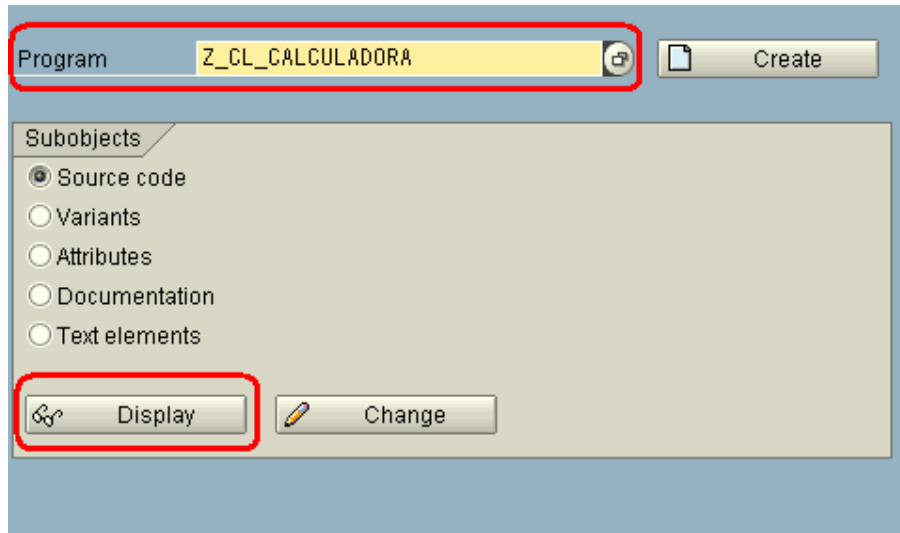
A transação SE38 é um programa de edição de REPORT, INCLUDE. Para criar um REPORT básico entre com os seguintes dados:

- Programa/Program: Nome do programa, os programas que não são desenvolvidos diretamente pela SAP (chamados de programas STANDARD), devem iniciar com a letra Z ou Y.
- Subobjetos: Texto fonte.

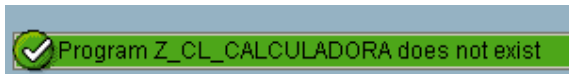


Criando uma Calculadora básica: Entre com o nome ZCALCULADORA00:

Primeiramente vamos criar um include que conterà a classe, um “INCLUDE” geralmente é um programa que contém parte ou trecho de código de um programa, neste include iremos criar a classe CL_CALCULADORA_OO, para exemplificar usaremos a nomenclatura Z_CL_CALCULADORA_OO para o include

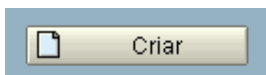


Clique em exibir e observe a barra de status:



Caso a mensagem seja esta “O programa Z_CL_CALCULADORA já existe” renomei o nome do programa, e repita os passos, caso contraria o programa pode ser criado com este nome.

Clique em criar:



ABAP: Program Attributes Z_CL_CALCULADORA Change

Title **Include da classe calculadora**

Original language EN English

Created 23.01.2008 CURS009

Last changed by

Status New(Revised)

Attributes

Type INCLUDE program

Status Test Program

Application

Authorization Group

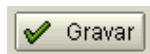
Editor lock

Save

A próxima tela é uma tela de configuração do INCLUDE.

- Título: Titulo do INCLUDE.
- Tipo: INCLUDE program

Clique em Gravar:



A próxima tela é referente a “criação de entrada de catalogo de objetos”, deixe o objeto salvo como local.

Criar entrada catálogo objetos

Objeto R3TR PROG ZCALCULADORA

Atributos

Pacote

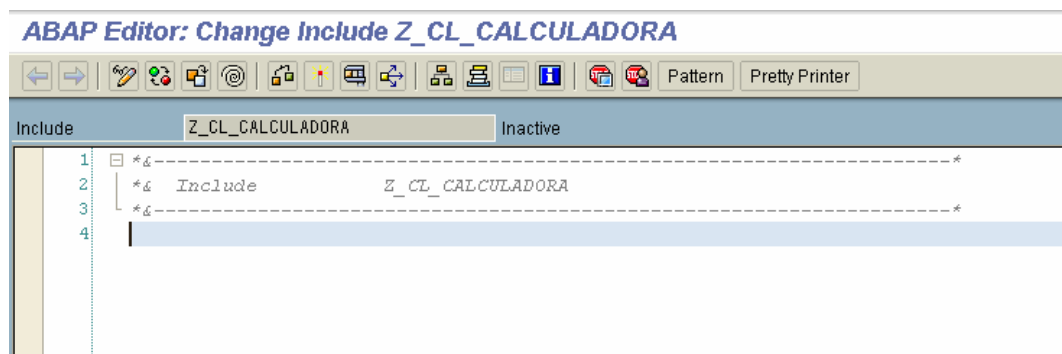
Responsável UDERSONFERM

Sistema de origem DEV

Idioma original PT Português

Objeto local Síntese bloqueio

Após a criação e configuração do INCLUDE será apresentado o editor ABAP para iniciar a codificação.



Neste Include será codificado a classe CL_CALCULADORA_OO, toda a lógica da classe será inserida neste include, segue a codificação e após a explicação:

```

1  | *-----
2  | | * Include           Z_CL_CALCULADORA
3  | | *-----
4  |
5  | *
6  | | * Definição da Classe
7  | | *
8  | | CLASS CL_CALCULADORA DEFINITION.
9  | |
10 | | *
11 | | * Sessão publica
12 | | *
13 | | PUBLIC SECTION.
14 | | *
15 | | * Variáveis do scope publico
16 | | *
17 | |     DATA: SINAL(1),
18 | |             VALORA TYPE I,
19 | |             VALORB TYPE I,
20 | |             SUBRC LIKE SY-SUBRC.
21 | |
22 | | *
23 | | * Métodos do scope publico
24 | | *
25 | |     METHODS CALC, DISPLAY.
26 | |
27 | | *
28 | | * Sessão privada
29 | | *
30 | | PRIVATE SECTION.
31 | |     DATA RESULT TYPE P DECIMALS 2.
32 | | ENDCLASS.
33 | |
34 | | *
35 | | * Implementação da classe CL_CALCULADORA
36 | | *
37 | |
38 | | CLASS CL_CALCULADORA IMPLEMENTATION.
39 | |
40 | | METHOD CALCULA.
41 | |     CLEAR: SUBRC, RESULT.
42 | |
43 | |     CASE SINAL.
44 | |         WHEN '+' .
45 | |             RESULT = VALORA + VALORB.
46 | |         WHEN '-' .
47 | |             RESULT = VALORA - VALORB.
48 | |         WHEN '/' .
49 | |             IF VALORB <> 0.
50 | |                 RESULT = VALORA / VALORB.
51 | |             ELSE.
52 | |                 SUBRC = SY-SUBRC.
53 | |             ENDIF.
54 | |         WHEN '*' .
55 | |             RESULT = VALORA * VALORB.
56 | |     ENDCASE.
57 | | ENDMETHOD.
58 | |     RESULT = VALORA * VALORB.
59 | |
60 | | METHOD DISPLAY.
61 | |     WRITE: 'Resultado:', RESULT.
62 | |
63 | | ENDMETHOD.
64 | | ENDCLASS.

```

Código:

```
*&-----
-*
*&  Include           Z_CL_CALCULADORA
*&-----
-*

*
* Definição da Classe
*
CLASS CL_CALCULADORA DEFINITION.

*
* Sessão publica
*
PUBLIC SECTION.
*
* Variaveis do scope publico
*
    DATA: SINAL(1),
           VALORA TYPE I,
           VALORB TYPE I,
           SUBRC LIKE SY-SUBRC.

*
* Métodos do scope publico
*
    METHODS: CALCULA, DISPLAY.

*
* Sessão privada
*
PRIVATE SECTION.
    DATA RESULT TYPE P DECIMALS 2.
ENDCLASS.

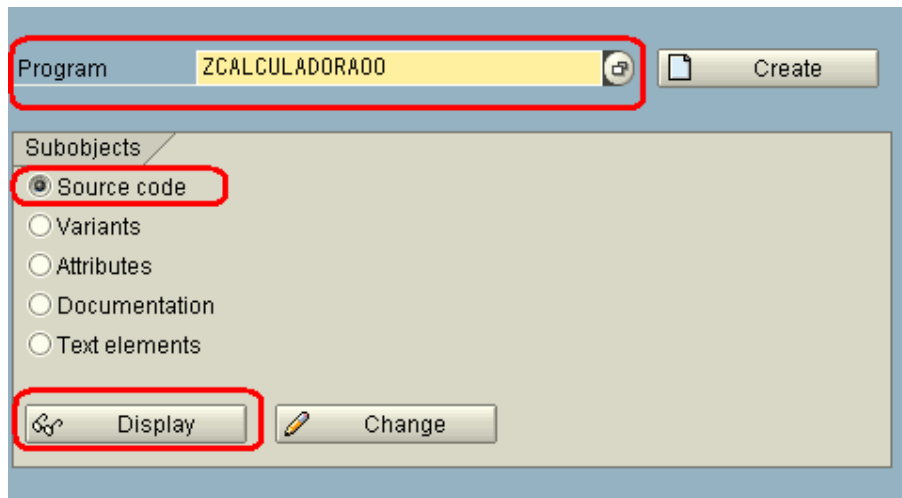
*
* Implementação da classe CL_CALCULADORA
*

CLASS CL_CALCULADORA IMPLEMENTATION.

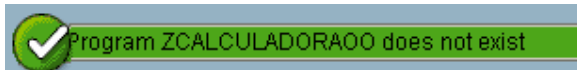
METHOD CALCULA.
    CLEAR: SUBRC, RESULT.

    CASE SINAL.
        WHEN '+' .
            RESULT = VALORA + VALORB.
        WHEN '-' .
            RESULT = VALORA - VALORB.
        WHEN '/' .
            IF VALORB <> 0.
                RESULT = VALORA / VALORB.
            ELSE .
                SUBRC = SY-SUBRC.
            ENDIF.
        WHEN '*' .
            RESULT = VALORA * VALORB.
    ENDCASE.
ENDMETHOD.
```

```
METHOD DISPLAY.  
    WRITE: 'Resultado:', RESULT.  
ENDMETHOD.  
ENDCLASS.
```

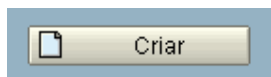


Clique em exibir e observe a barra de status:



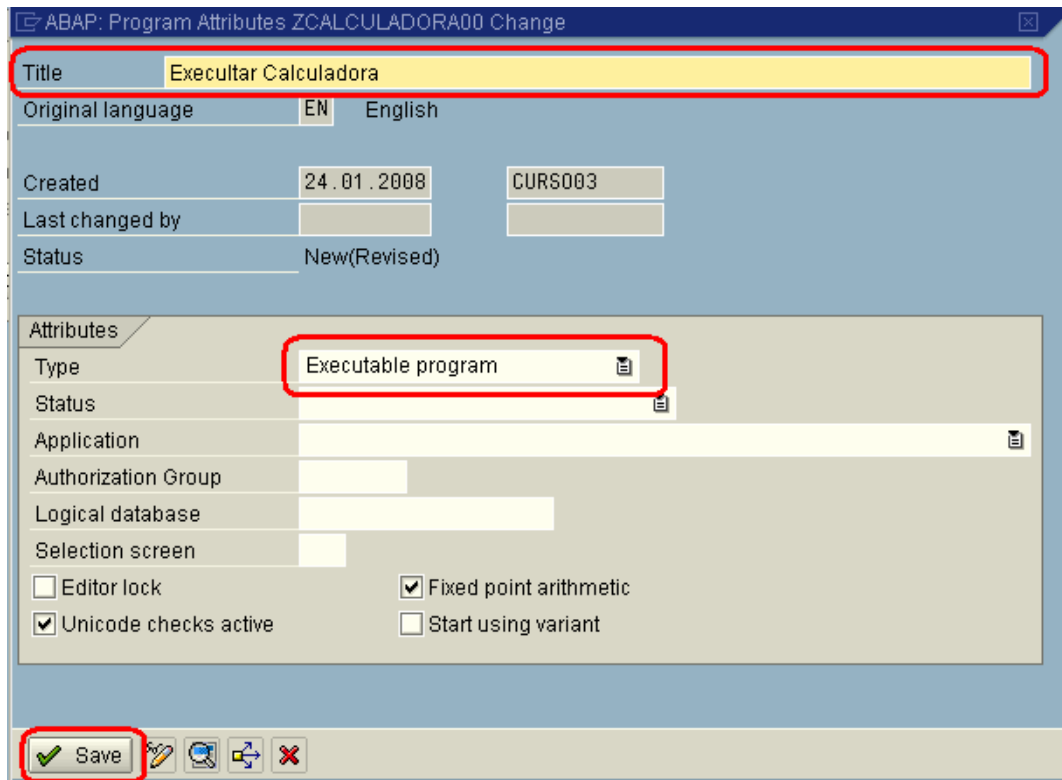
Caso a mensagem seja esta “O programa ZCALCULADORA já existe” renomei o nome do programa, e repita os passos, caso contraria o programa pode ser criado com este nome.

Clique em criar:

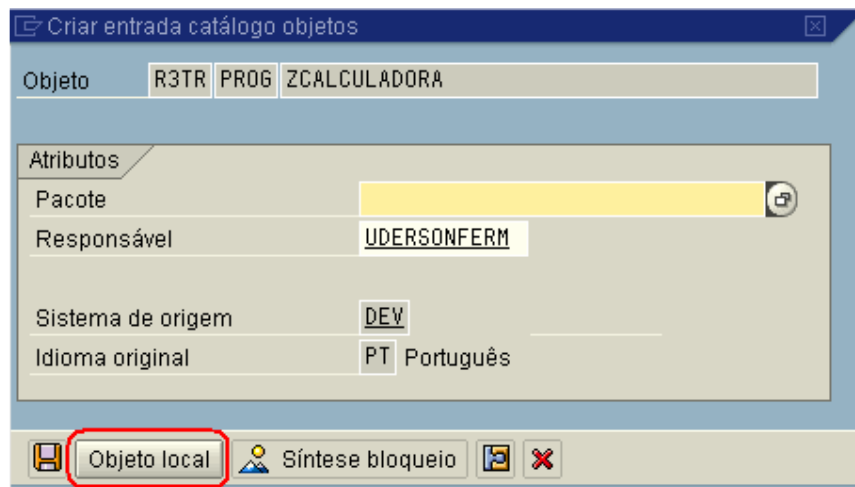


A próxima tela é uma tela de configuração do REPORT.

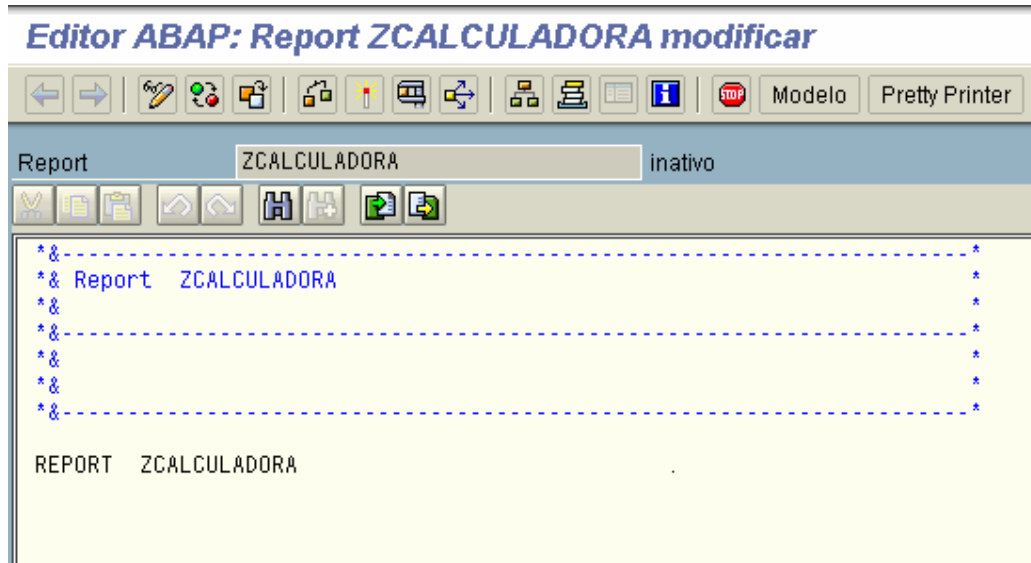
- Título: Titulo do REPORT.
- Tipo: tipo do programa, geralmente e neste caso “Programa Executável”.



A próxima tela é referente a “criação de entrada de catálogo de objetos”, deixe o objeto salvo como local.



Após a criação e configuração do REPORT será apresentado o editor ABAP para iniciar a codificação.



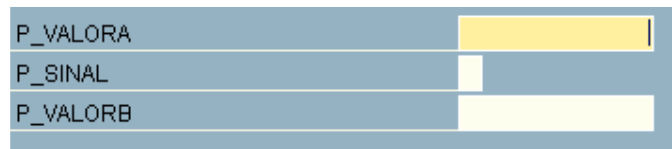
Edite a lógica da calculadora:

```
1  *&-----*
2  *& Report  ZCALCULADORA00
3  *&
4  *&-----*
5  *&
6  *&
7  *&-----*
8
9  REPORT  ZCALCULADORA00.
10
11  INCLUDE Z_CL_CALCULADORA.
12
13  DATA CALCULADORA TYPE REF TO CL_CALCULADORA.
14
15  PARAMETER: VALORA TYPE I,
16             SINAL  TYPE C,
17             VALORB TYPE I.
18
19  START-OF-SELECTION.
20      CREATE OBJECT CALCULADORA.
21
22      CALCULADORA->VALORA = VALORA .
23      CALCULADORA->VALORB = VALORB.
24      CALCULADORA->SINAL  = SINAL.
25
26      CALL METHOD CALCULADORA->CALCULA.
27
28  IF CALCULADORA->SUBRC EQ 0.
29      CALL METHOD CALCULADORA->DISPLAY.
30  ENDIF.
```

Entendendo:

- REPORT <..nome do report...>.
- INCLUDE <...nome_do_include...>: inclui um include dentro do report neste momento o REPORT consegue coletar todas as informações que estão dentro do INCLUDE.
- DATA <...nome_da_variavel...> TYPE REF TO<...objeto..> declaração de uma referencia para um objeto.
 - CL_CALCULADORA que é a classe que foi criada dentro do include.
- DATA <...nome_da_variavel...> TYPE <...tipo de dados..>
 - I = tipo de dado inteiro sem casa decimal
 - C = tipo caracter ASCII
- PARAMETER < ...nome_do_paramenter ...> TYPE <... tipo de dado ...>

Este componente gera uma saída de entrada de dado pode ser comparado por uma componente de entrada de texto, onde o comando TYPE é que define o tipo de dado que será aceito.



- **START-OF-SELECTION.** Este commando é acionando antes de executar qualquer parte do REPORT geralmente usado para iniciar variáveis, objeto.
- **CREATE OBJECT** CALCULADORA. Cria (Instancia) o objeto CALCULADORA que foi declarado no comando DATA ... TYPE REF TO
- OBJETO->Variável insere ou retira valores das variáveis que são do escopo publicas:

```
CALCULADORA->VALORA = VALORA .  
CALCULADORA->VALORB = VALORB.  
CALCULADORA->SINAL = SINAL.
```

Após editar este programa, é necessário primeiro mente seguir os três passos básicos para qualquer tipo de programação dentro SAP.

- Após setar os valores das variáveis foi chamado o método CALCULA, que fará os cálculos sobre os valores das variáveis.

CALL METHOD CALCULADORA->CALCULA.

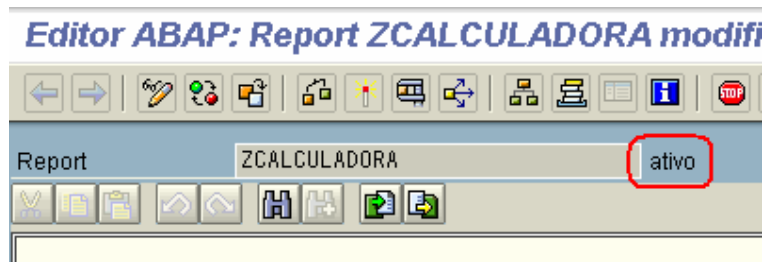
- Se o valor da variável SUBRC estive com zero (0), será acionado o método DISPLAY que imprimira os valores das variáveis.

○ SALVAR = 

○ CKECAR = 

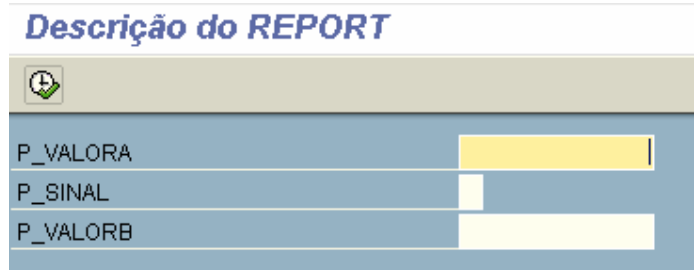
○ ATIVAR = 

Após seguir estas etapas observe o status do REPORT:

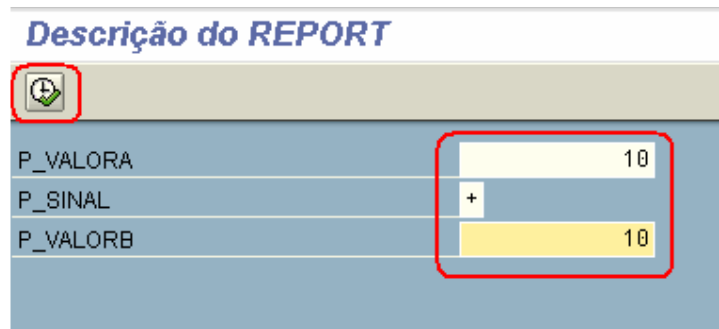


Caso esteja já ativo basta executar o programa:

 ou Tecla F8

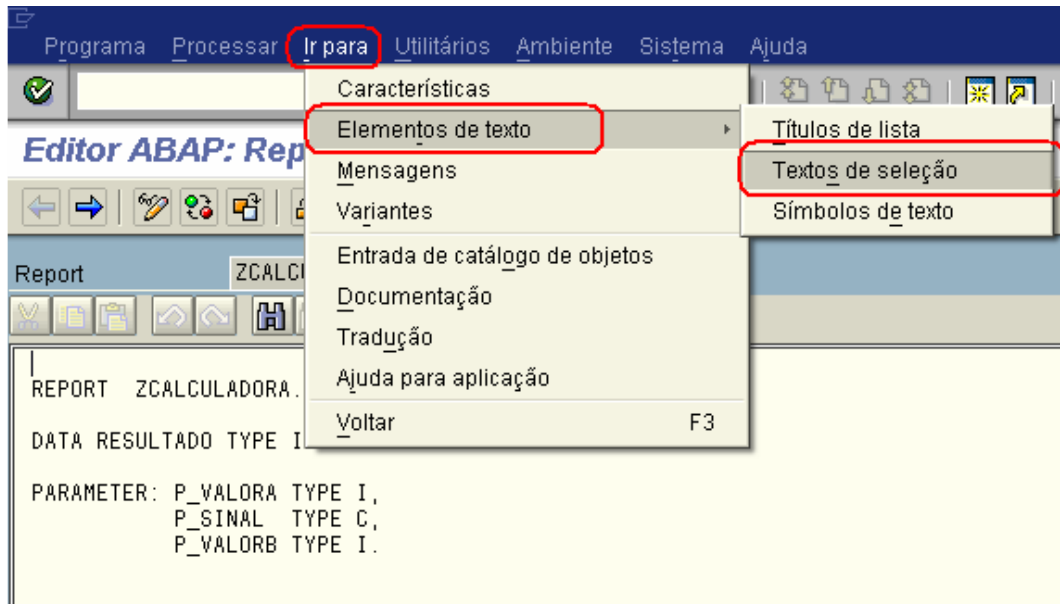


Entre com os dados:

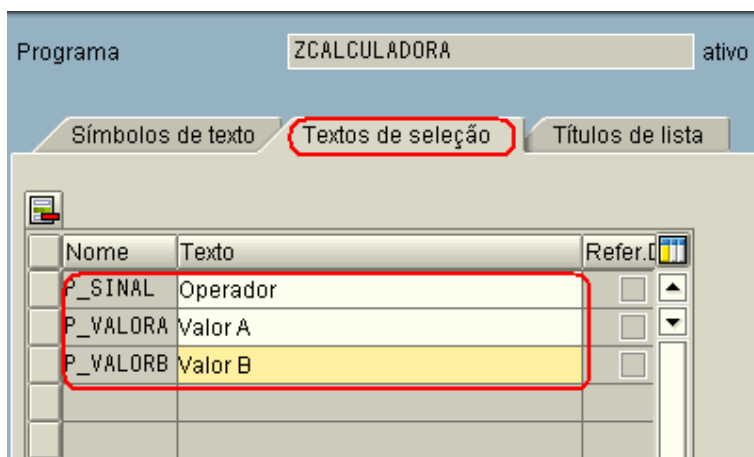






Descrição do REPORT
Descrição do REPORT
20

Observe que os parameter estão aparecendo com os próprios nomes, para que ele possa aparecer com label definidas pelo usuário, basta seguir:



Entre com os TEXTOS de SELEÇÃO, para cara PARAMETER.



- SALVE = 
- ATIVE = 
- VOLTE = 
- EXECULTE = 

Observe agora:

