Class: ZCL_FI_BUDGET_ROOT

Status: Active

Attributes

Description: Budget root class for shared memory
Instantiation: Public
Final

Not released
Fixed pt.arithmetic
Category: General Object Type
Package: ZBRFI
Original lang.: EN

Created by: ABAP
Created on: 09.04.2014
Last changed by: ABAP
Last changed on: 09.04.2014

Attribute

Private attribute

Attrib.	Cat	Description	Ref. Type	Init. value
AT_RETURN	Inst	Error Messages	TYPE BAPIRET2_TAB	

Methods

Public methods

GET_MY_BUDGET_MESSAGES
Description: Returns the budget messages addressed to current user.
Instance mthd

Returning parameter
  VALUE(RT_MESSAGES) TYPE BAPIRET2_TAB (Error Messages)

```
method get_my_budget_messages.
  field-symbols <lw_return> type bapiret2.
  loop at me->at_return assigning <lw_return> where system = sy-uname.
    append <lw_return> to rt_messages.
    DELETE me->at_return.
  endloop.
endmethod.
```

SET_BUDGET_MESSAGES
Description: Sets the messages table.
Instance mthd

Importing parameter
  VALUE(PT_RETURN) TYPE BAPIRET2_TAB (Error Messages)

```
method SET_BUDGET_MESSAGES.
  data: lv_timst TYPE timestamp,
```

```
        lv_ttims TYPE timestamp,
        lv_difer TYPE int4.
  FIELD-SYMBOLS <lw_return> TYPE bapiret2.
  get TIME STAMP FIELD lv_timst.
  LOOP AT me->at_return ASSIGNING <lw_return>.
    lv_ttims = <lw_return>-PARAMETER.
    lv_difer = lv_timst - lv_ttims.
    IF lv_difer > 140.
      delete me->at_return.
    ENDIF.
  ENDLOOP.
  LOOP AT pt_return ASSIGNING <lw_return>.
    <lw_return>-PARAMETER = lv_timst.
    <lw_return>-system    = sy-uname.
    APPEND <lw_return> to me->at_return.
  ENDLOOP.
endmethod.
```

Redefined Methods

Local Types

```
*"* use this source file for any type of declarations (class
*"* definitions, interfaces or type declarations) you need for
*"* components in the private section
```

Local class definitions

```
*"* use this source file for the definition and implementation of
*"* local helper classes, interface definitions and type
*"* declarations
```

Macros

```
*"* use this source file for any macro definitions you need
*"* in the implementation part of the class
```

Overview