

```

*&-----*
*& Report Z_Simpletree *
*& Author : Swarna.S. *
*&-----*
*& AS : Simple tree report using RS_TREE_CONSTRUCT *
*& and RS_TREE_LIST_DISPLAY upto 6 levels *
*&-----*

REPORT z_simpletree.

* type pool declarations for tree
TYPE-POOLS : fibs, stree.

*Data declaration for additional node information
DATA : t_node TYPE snodetext.

*Internal table and wa decl for nodes
DATA : it_node LIKE TABLE OF t_node INITIAL SIZE 0,
       wa_node LIKE t_node.

*Internal table and wa decl for Education table PA0022
DATA : it_0022 TYPE STANDARD TABLE OF pa0022 INITIAL SIZE 0,
       wa_0022 TYPE pa0022.

*Internal table and wa decl for text table t517x
DATA : it_517x TYPE STANDARD TABLE OF t517x INITIAL SIZE 0,
       wa_517x TYPE t517x.

*Internal table and wa decl for text table t517T
DATA : it_517t TYPE STANDARD TABLE OF t517t INITIAL SIZE 0,
       wa_517t TYPE t517t.

*Internal table and wa decl for text table t519T
DATA : it_519t TYPE STANDARD TABLE OF t519t INITIAL SIZE 0,
       wa_519t TYPE t519t.

*initialization event
INITIALIZATION.

*Start of selection event
START-OF-SELECTION.

*Select the data for tree
PERFORM fetch_data.

*Build the hierarchy for tree
PERFORM build_hierarchy.

*Build Tree for display
PERFORM build_tree.

*&-----*
*& Form fetch_data *
*&-----*
* text *
*&-----*
* --> p1 text *
* <-- p2 text *
*&-----*
FORM fetch_data .

*select data from PA0022

```

```
SELECT * FROM pa0022 INTO CORRESPONDING FIELDS OF TABLE it_0022 UP TO 50
ROWS.
```

```
*select data from T517x
```

```
SELECT * FROM t517x INTO CORRESPONDING FIELDS OF TABLE it_517x WHERE langu =
'E'.
```

```
*select data from T517T
```

```
SELECT * FROM t517t INTO CORRESPONDING FIELDS OF TABLE it_517t WHERE sprsl =
'E'.
```

```
*select data from T519t
```

```
SELECT * FROM t519t INTO CORRESPONDING FIELDS OF TABLE it_519t WHERE sprsl =
'E'.
```

```
ENDFORM.                                " fetch_data
*&-----*
*&      Form  build_hierarchy
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM build_hierarchy .
```

```
*Building the nodes and hierarchy for tree
```

```
CLEAR : it_node[], wa_node.
```

```
wa_node-type = 'T'.
wa_node-name = 'Education'.
wa_node-tlevel = '01'.
wa_node-nlength = '15'.
wa_node-color = '4'.
wa_node-text = 'Infotype 0022'.
wa_node-tlength = '20'.
wa_node-tcolor = 3.
APPEND wa_node TO it_node.
CLEAR wa_node.
```

```
*Filling the values of internal table into tree
```

```
LOOP AT it_0022 INTO wa_0022.
```

```
wa_node-type = 'P'.
wa_node-name = 'PERNR'.
wa_node-tlevel = '02'.
wa_node-nlength = '8'.
wa_node-color = '1'.
wa_node-text = wa_0022-pernr.
wa_node-tlength = '20'.
wa_node-tcolor = 4.
APPEND wa_node TO it_node.
CLEAR wa_node.
```

```
*Filling the text of T517t
```

```
READ TABLE it_517t INTO wa_517t WITH KEY slart = wa_0022-slart.
```

```
wa_node-type = 'P'.
wa_node-name = wa_0022-slart.
wa_node-tlevel = '03'.
wa_node-nlength = '8'.
```

```
wa_node-color = '1'.
wa_node-text = wa_517t-stext.
wa_node-tlength = '40'.
wa_node-tcolor = 4.
APPEND wa_node TO it_node.
CLEAR wa_node.
```

***Filling the text of T519t**

```
READ TABLE it_519t INTO wa_519t WITH KEY slabs = wa_0022-slabs.
```

```
wa_node-type = 'P'.
wa_node-name = wa_0022-slabs.
wa_node-tlevel = '04'.
wa_node-nlength = '8'.
wa_node-color = '2'.
wa_node-text = wa_519t-stext.
wa_node-tlength = '40'.
wa_node-tcolor = 4.
APPEND wa_node TO it_node.
CLEAR wa_node.
```

***Filling the text of T517x**

```
READ TABLE it_517x INTO wa_517x WITH KEY faart = wa_0022-sltpl.
```

```
wa_node-type = 'P'.
wa_node-name = wa_0022-sltpl.
wa_node-tlevel = '05'.
wa_node-nlength = '8'.
wa_node-color = '1'.
wa_node-text = wa_517x-ftext.
wa_node-tlength = '40'.
wa_node-tcolor = 4.
APPEND wa_node TO it_node.
CLEAR wa_node.
```

```
wa_node-type = 'P'.
wa_node-name = wa_0022-zzper.
wa_node-tlevel = '06'.
wa_node-nlength = '8'.
wa_node-color = '1'.
wa_node-text = '% Completed'.
wa_node-tlength = '15'.
wa_node-tcolor = 4.
APPEND wa_node TO it_node.
CLEAR wa_node.
```

```
ENDLOOP.
```

```
ENDFORM.                                " build_hierarchy
*&-----*
*&      Form      build_tree
*&-----*
*      text
*-----*
* -->  p1      text
* <--  p2      text
*-----*
FORM build_tree .
```

***Fm for constructing the tree**

```
CALL FUNCTION 'RS_TREE_CONSTRUCT'
  TABLES
    nodetab = it_node.
```

```

IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*   WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

*FM for displaying the tree
CALL FUNCTION 'RS_TREE_LIST_DISPLAY'
  EXPORTING
    callback_program      = sy-repid
    check_duplicate_name = '1'
    color_of_node         = '4'
    color_of_mark        = '3'
    color_of_link         = '1'
    color_of_match        = '5'
    node_length           = 30
    text_length           = 75
    use_control           = 'L'.

ENDFORM.                " build_tree

```

Output

