

Web Dynpro for ABAP: Usage of Complex Restrictions in Select Options



Applies to:

This document applies to SAP ECC 6.0, SAP Netweaver 2004s. For more information, visit the [Web Dynpro ABAP homepage](#).

Summary

This article guides you on the usage of complex restrictions in select options in WDA.

Author: Susheela Chandralekha Bommakanti

Company: Yash Technologies Pvt. Ltd.

Created on: 26 April 2011

Author Bio

Susheela Chandralekha Bommakanti is working as a SAP Technical Consultant in Yash Technologies Pvt. Ltd. She has more than 3 years of experience in SAP.

Table of Contents

Introduction	3
Step by Step Process	3
Step 1: Create a Web Dynpro Component	3
Step 2: Add Component usage	4
Step 3: Design the layout	5
Step 4: Embedding a view in a window	5
Step 5: Coding	6
Step 6: Create Application	8
Application – Output	8
Test case #1	8
Test case #2	8
Test case #3	9
Related Content	10
Disclaimer and Liability Notice	11

Introduction

This is a demo on how to use the complex restrictions in Select options in Web Dynpro for ABAP.

This will guide you on different scenarios or test cases of using the complex restrictions in select options.

Sometimes, there will be lot of space constraints when using the select option. Using these complex restrictions we can do it and also limit the user selections based on this.

Step by Step Process

Step 1: Create a Web Dynpro Component

Add a component usage for select option at component level

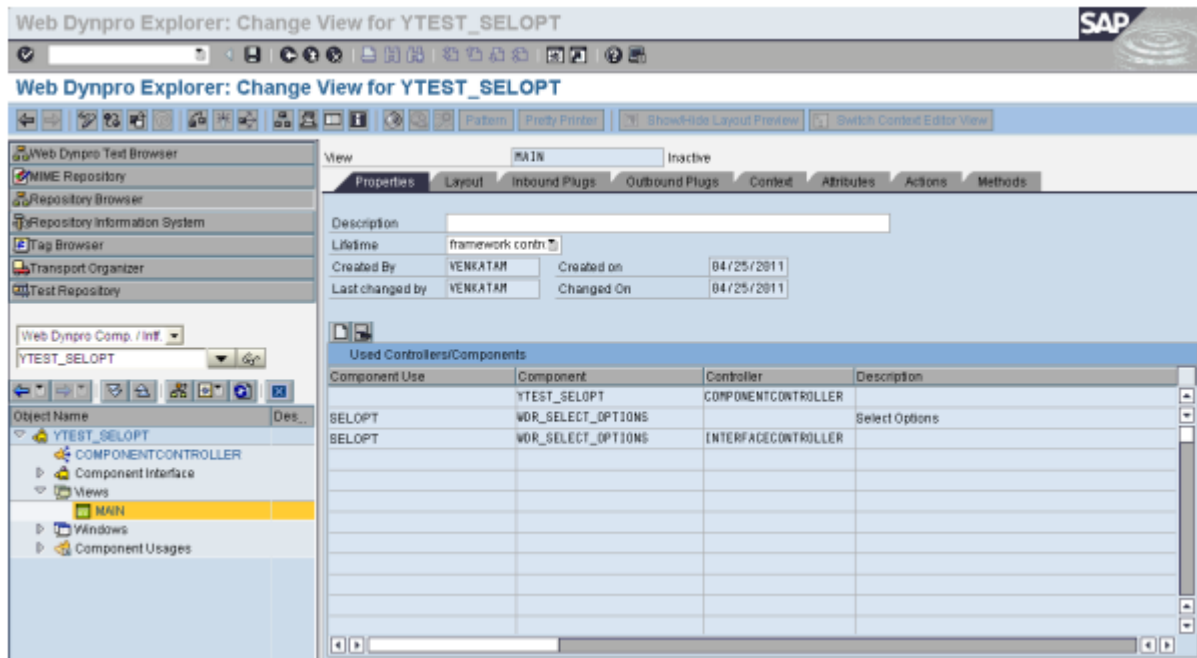
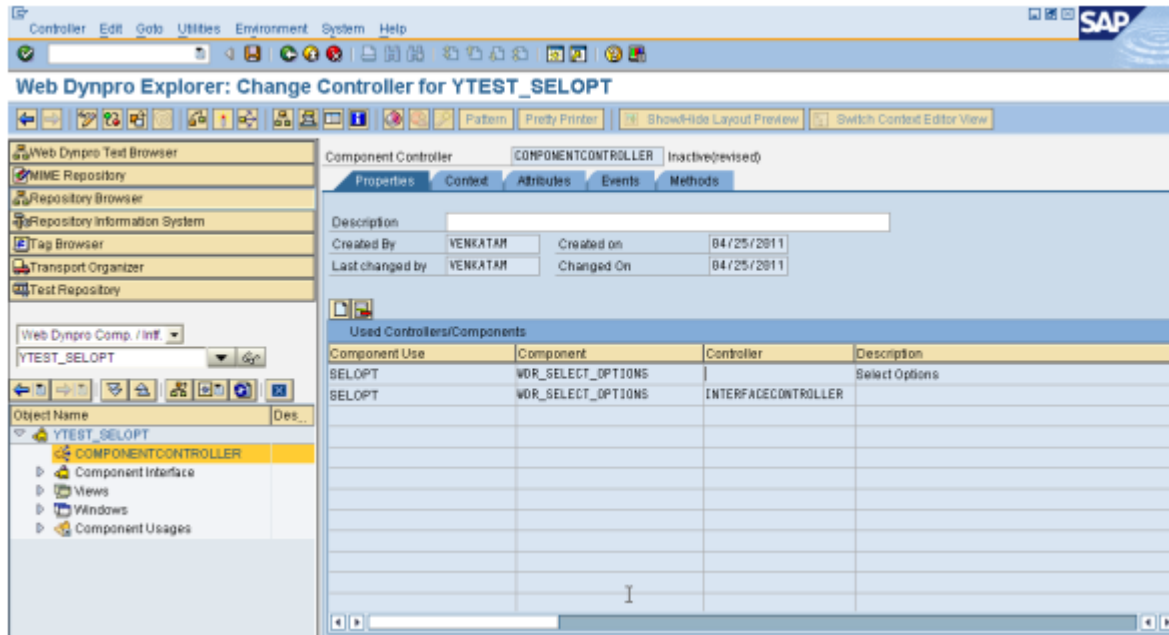
The screenshot shows the SAP Web Dynpro Explorer interface for a component named 'YTEST_SELOPT'. The component is marked as 'Inactive/Revised'. The 'Used Components' tab is selected, displaying a table with the following data:

Component Use	Component	Description of Component
SELOPT	wDR_SELECT_OPTIONS	Select Options

Step 2: Add Component usage

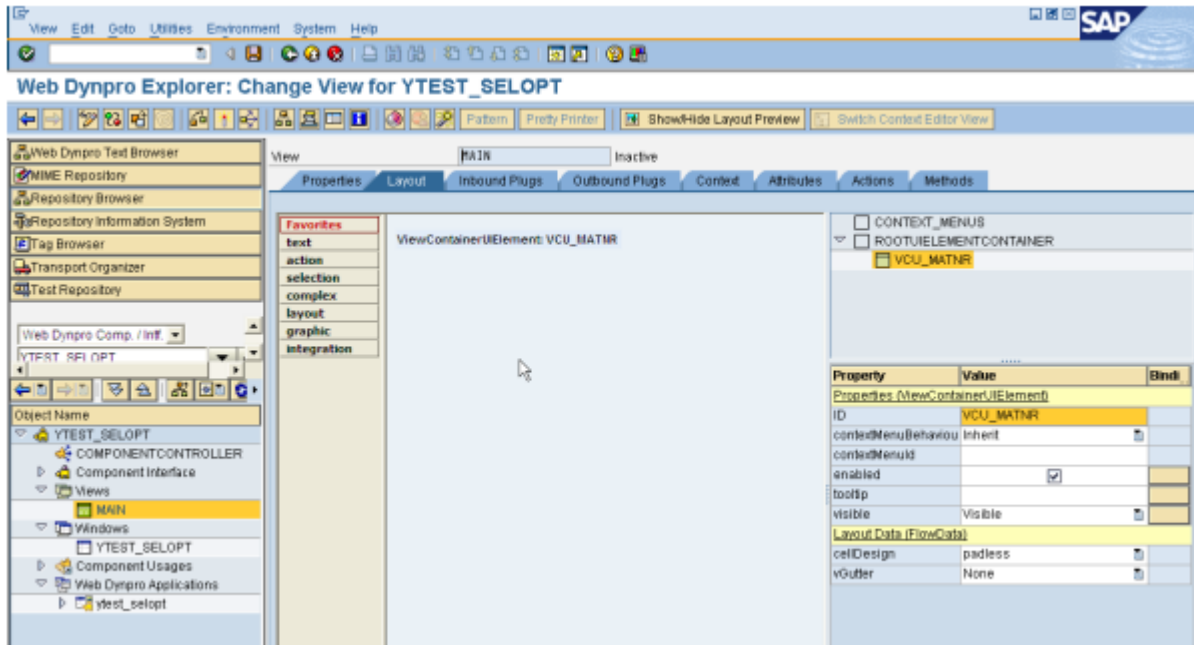
Add a component usage in both Component controller and view.

Add the usage in both component controller and MAIN view as shown below –



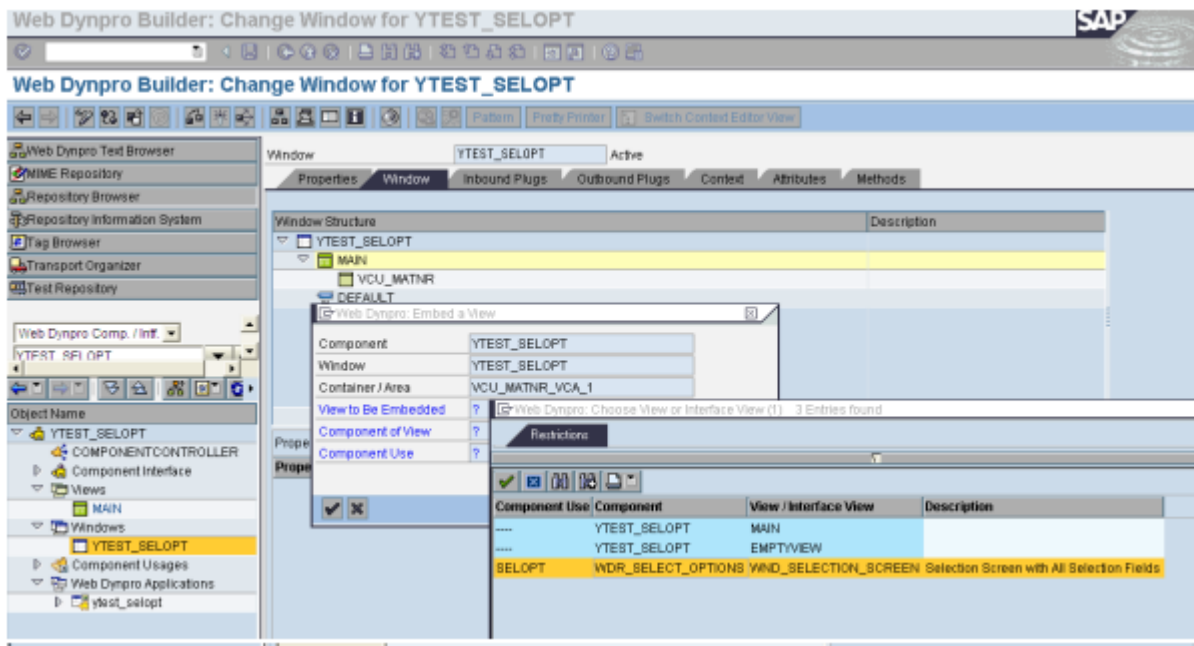
Step 3: Design the layout

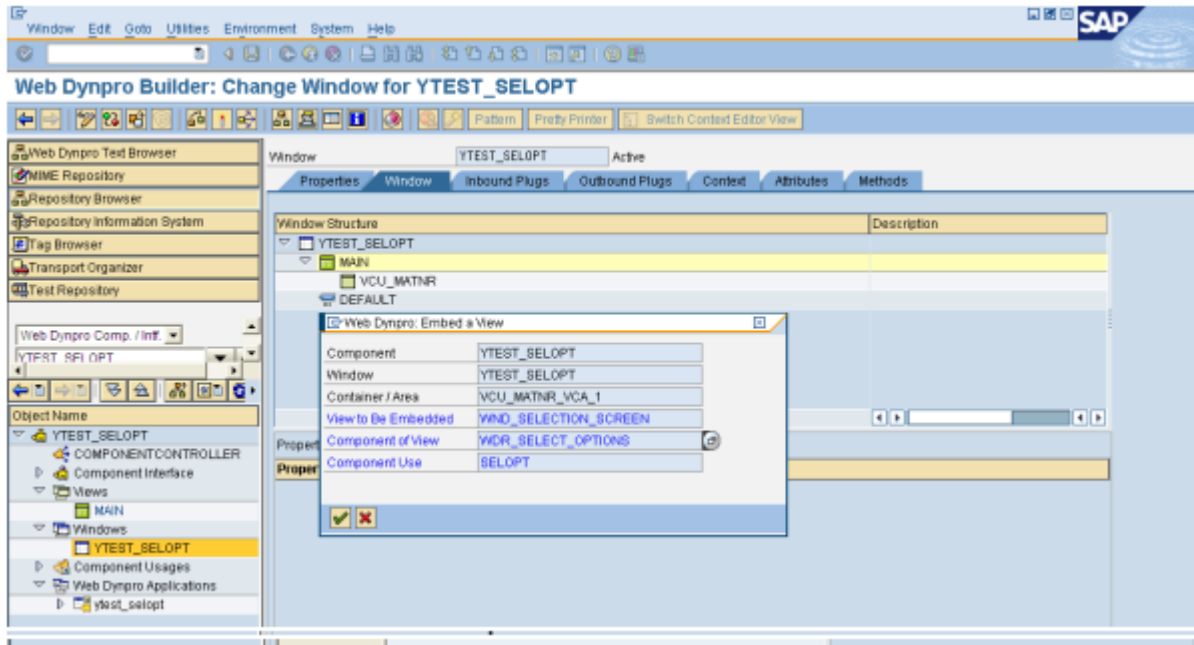
Create a view container element in the MAIN view as shown below –



Step 4: Embedding a view in a window

Embed the view for view container element by selecting the particular view of a component usage as shown below –





Step 5: Coding

Using the code wizard, instantiate the component usage in the WDDOINIT of the MAIN view as per the below code.

```
METHOD wddoinit .
  DATA:
    lo_cmp_usage          TYPE REF TO if_wd_component_usage,
    lo_interfacecontroller TYPE REF TO iwci_wdr_select_options,
    lo_selopt             TYPE REF TO if_wd_select_options,
    rt_matnr             TYPE REF TO data,
    ls_complex_restrictions TYPE if_wd_select_options=>t_complex_restrictions.

  * Create the instance
  lo_cmp_usage = wd_this->wd_cpuse_selopt( ).
  IF lo_cmp_usage->has_active_component( ) IS INITIAL.
    lo_cmp_usage->create_component( ).
  ENDIF.
  lo_interfacecontroller = wd_this->wd_cpifc_selopt( ).
  * Handler class
  IF lo_interfacecontroller IS NOT INITIAL.
    lo_selopt = lo_interfacecontroller->init_selection_screen(
    ).
  ENDIF.
  * Create range table
  IF lo_selopt IS NOT INITIAL.
    CALL METHOD lo_selopt->create_range_table
      EXPORTING
        i_typename      = 'MATNR'
      RECEIVING
        rt_range_table = rt_matnr.
  ENDIF.
  * Remove Global settings -
  CALL METHOD lo_selopt->set_global_options
    EXPORTING
      i_display_btn_cancel = abap_false
```

```

i_display_btn_check   = abap_false
i_display_btn_reset   = abap_false
i_display_btn_execute = abap_false.

* Usage of complex restrictions
* -----TEST CASE #1-----*
* Range related parameters are not passed.
* Either do not pass any parameters for ranges like NB, BT
* OR Pass SPACE to NB and BT. both will give same output.
* -----*
*   ls_complex_restrictions-m_include-bt = space.
*   ls_complex_restrictions-m_include-nb = space.
*   ls_complex_restrictions-m_include-lt = abap_true.
*   ls_complex_restrictions-m_include-le = abap_true.
*   ls_complex_restrictions-m_include-gt = abap_true.
*   ls_complex_restrictions-m_include-ge = abap_true.
*   ls_complex_restrictions-m_include-eq = abap_true.
*   ls_complex_restrictions-m_include-cp = abap_true.
*   ls_complex_restrictions-m_include-np = abap_true.
*   ls_complex_restrictions-m_include-ne = abap_true.

* -----* TEST CASE #2-----*
* All parameters are passed
* -----*
*   ls_complex_restrictions-m_include-bt = abap_true.
*   ls_complex_restrictions-m_include-nb = abap_true.
*   ls_complex_restrictions-m_include-LT = abap_true.
*   ls_complex_restrictions-m_include-LE = abap_true.
*   ls_complex_restrictions-m_include-GT = abap_true.
*   ls_complex_restrictions-m_include-GE = abap_true.
*   ls_complex_restrictions-m_include-EQ = abap_true.
*   ls_complex_restrictions-m_include-CP = abap_true.
*   ls_complex_restrictions-m_include-NP = abap_true.
*   ls_complex_restrictions-m_include-NE = abap_true.

* -----TEST CASE #3-----*
* EXCLUDE only LESS THAN option
* -----*
*   ls_complex_restrictions-m_include-bt = abap_true.
*   ls_complex_restrictions-m_include-nb = abap_true.
*   ls_complex_restrictions-m_include-LT = space.
*   ls_complex_restrictions-m_include-LE = abap_true.
*   ls_complex_restrictions-m_include-GT = abap_true.
*   ls_complex_restrictions-m_include-GE = abap_true.
*   ls_complex_restrictions-m_include-EQ = abap_true.
*   ls_complex_restrictions-m_include-CP = abap_true.
*   ls_complex_restrictions-m_include-NP = abap_true.
*   ls_complex_restrictions-m_include-NE = abap_true.

* Add selection field for material number
CALL METHOD lo_selopt->add_selection_field
EXPORTING
  i_id           = 'ID_MATNR'
  it_result      = rt_matnr
  i_complex_restrictions = ls_complex_restrictions
  i_use_complex_restriction = abap_true

```

```

i_no_extension      = abap_false
i_no_intervals      = abap_true.
ENDMETHOD.

```

Step 6: Create Application

Save and activate the component and create an application for the component.

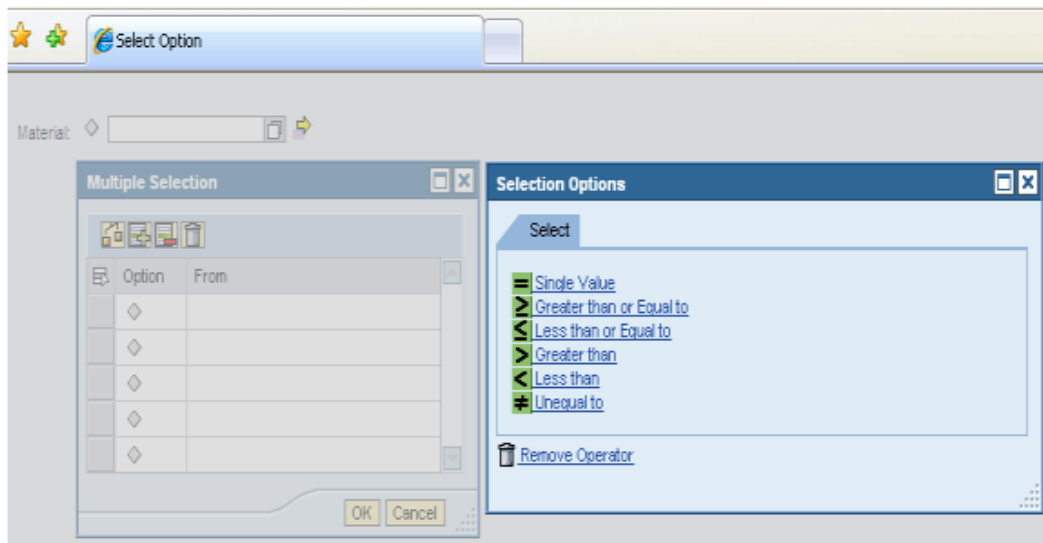
Application – Output

In Coding, different test cases are provided and their output for each case is shown below –

In all test cases, Intervals is not shown either in MAIN view. Only Extension Icon is shown.

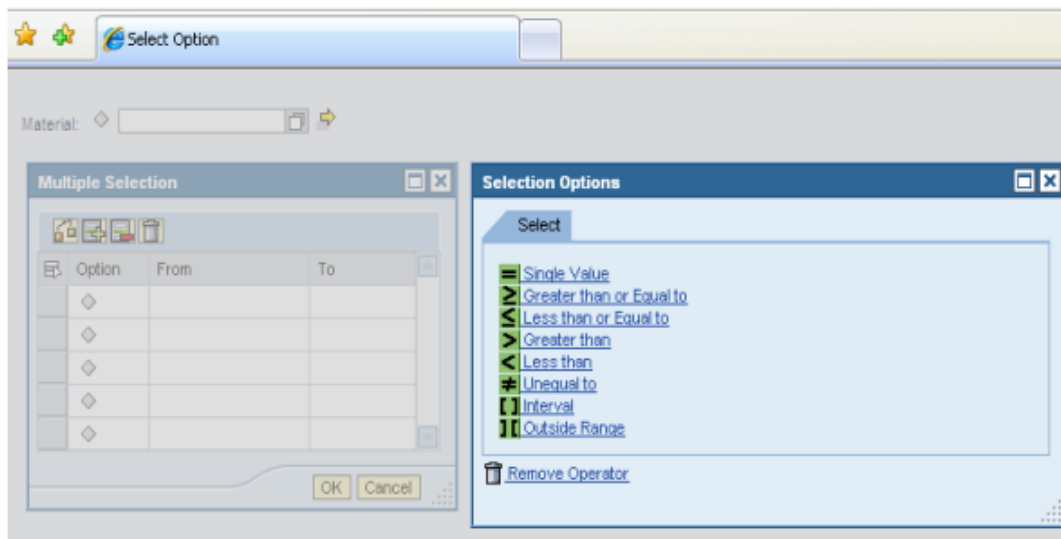
Test case #1

'TO' option is not shown in multiple selection popup. Only 'FROM' option is shown and respective selection operators are shown.



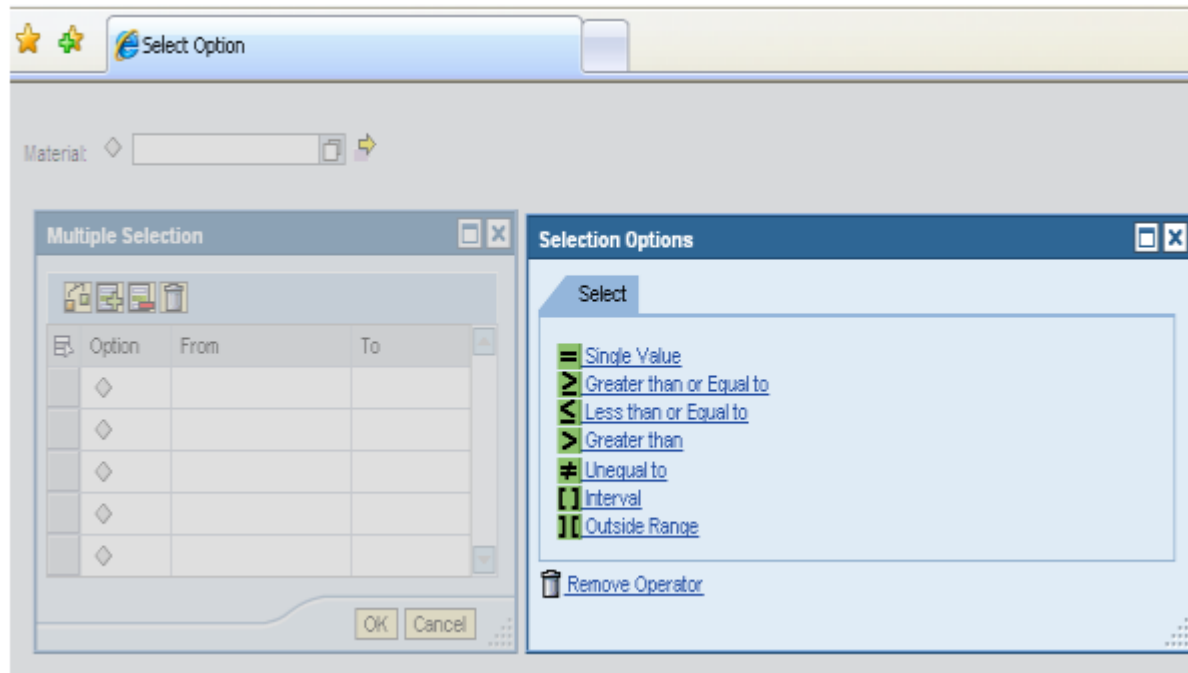
Test case #2

Both 'FROM and TO' is shown in multiple selection popup. This is a generic case where all selection operators are shown.



Test case #3

Both 'FROM and TO' is shown in multiple selection popup. Other than **Less Than** operator, all others are shown.



As per the requirement we can handle the select options using complex restrictions.

Related Content

[WDA Tutorial I: Getting Started with Web Dynpro for ABAP.](#)

[WDA Tutorial II: Getting Started with Web Dynpro for ABAP.](#)

For more information, visit the [Web Dynpro ABAP homepage](#).

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.