

# Handling Table Control in eCATT - Demo.



## Applies to:

SAP ECC 6.0. For more information, visit the [Business Intelligence homepage](#).

## Summary

Demo to handle table controls while using eCATT is explained in this article.

**Author:** Ashok Kumar Rajagopal

**Company:** Steria India Limited.

**Created on:** 14 DEC 2009.

## Authors Bio



I am Ashok Kumar Rajagopal, One of the Senior SAP ABAP Consultant in our company Steria India Limited. Currently I am playing the role of Team Leader. I have worked in USA for around 4 years and also worked in client place located in Amsterdam, Netherlands.

## Table of Contents

Creating Test Script.....	3
Record creation of purchase order.....	6
Step 1:.....	6
Step 2:.....	6
Step 3:Line Item no.10.....	6
Step 4:.....	6
Step 5:.....	6
Parameterization of the script.....	8
Step 1:.....	8
Step 2:.....	8
Step 3:.....	8
Step 4 :.....	9
Step 5:.....	9
Testing the Script:.....	10
Pre-request for testing.....	10
Testing the script as it is.....	10
Test Data:.....	10
Expected Result.....	10
Actual result.....	10
Modification of the recording of Test Script.....	11
Parameterize the control ID of Material in the recording ME21N_14_STEP_4.....	11
Testing after modification of Script.....	12
Conclusion:.....	12
Related Content.....	13
Disclaimer and Liability Notice.....	14

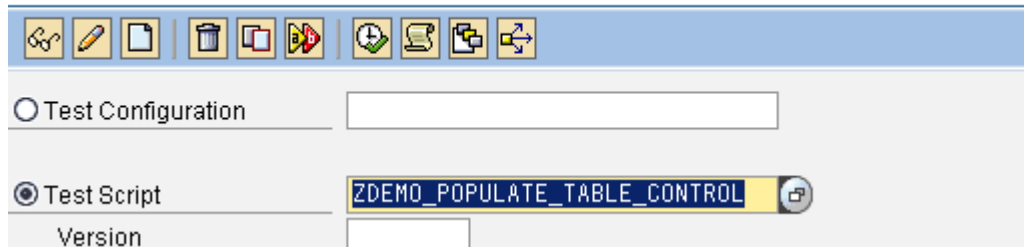
## Introduction

The intention of this article is to provide step by step approach for handling table controls while using the eCATT.

## Creating Test Script.

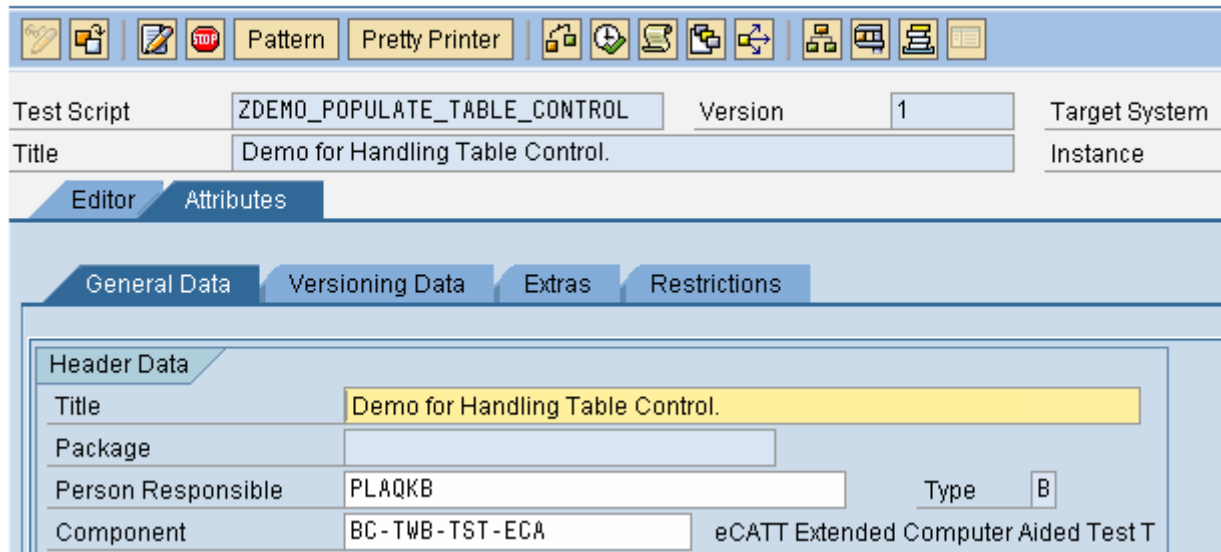
Through transaction SECATT , create test script ZDEMO\_POPULATE\_TABLE\_CONTROL.

### Extended Computer Aided Test Tool: Initial Screen



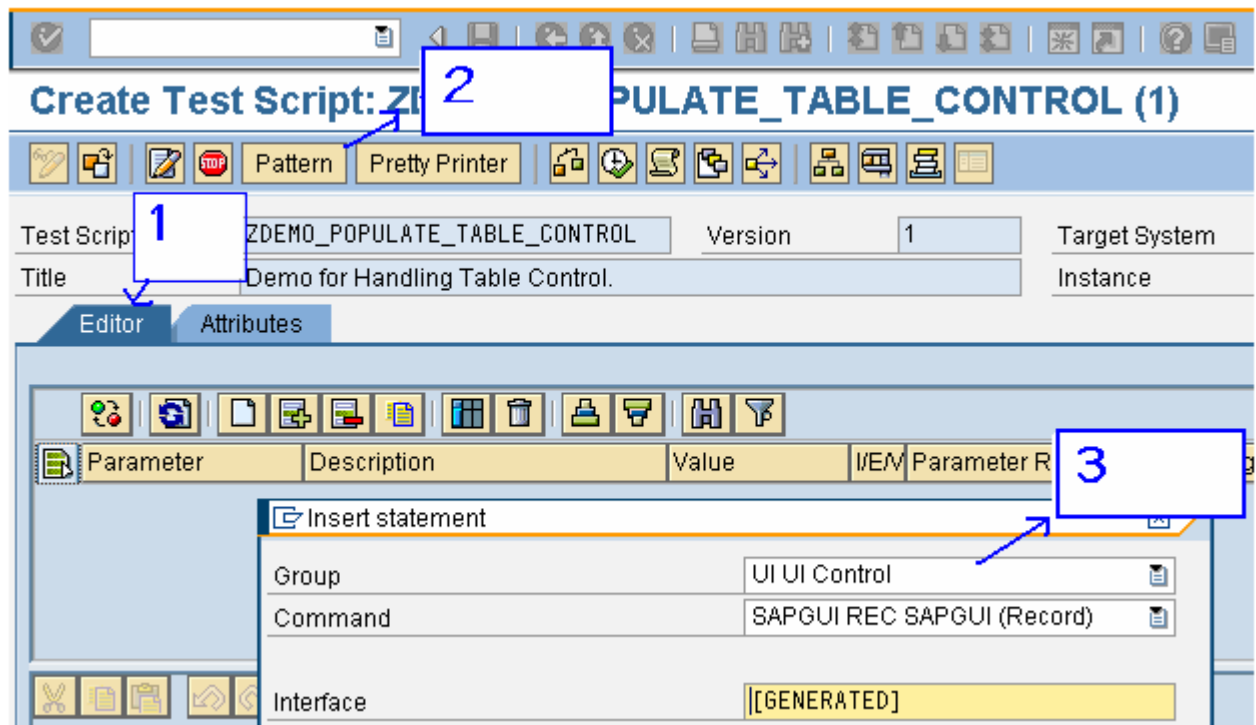
The screenshot shows the initial screen of the Extended Computer Aided Test Tool. It features a toolbar at the top with various icons. Below the toolbar, there are two main sections. The first section is labeled 'Test Configuration' and contains an empty text input field. The second section is labeled 'Test Script' and is selected with a radio button. It contains a text input field with the value 'ZDEMO\_POPULATE\_TABLE\_CONTROL' and a small icon to its right. Below this, there is a 'Version' label followed by an empty text input field.

### Create Test Script: ZDEMO\_POPULATE\_TABLE\_CONTROL (1)

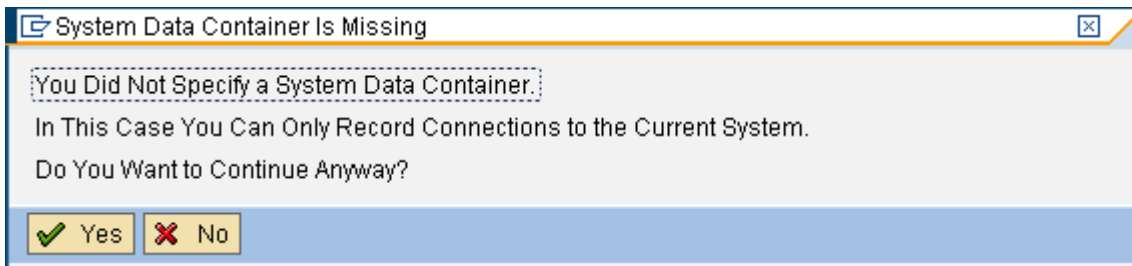


The screenshot shows the 'Create Test Script' dialog box in eCATT. The dialog has a toolbar at the top with icons for 'Pattern', 'Pretty Printer', and other functions. Below the toolbar, there are several input fields and tabs. The 'Test Script' field contains 'ZDEMO\_POPULATE\_TABLE\_CONTROL', the 'Version' field contains '1', and the 'Target System' field is empty. The 'Title' field contains 'Demo for Handling Table Control.' and the 'Instance' field is empty. There are two main tabs: 'Editor' and 'Attributes'. The 'Attributes' tab is selected, and it has four sub-tabs: 'General Data', 'Versioning Data', 'Extras', and 'Restrictions'. The 'General Data' sub-tab is selected, and it shows a 'Header Data' section with the following fields: 'Title' (Demo for Handling Table Control.), 'Package' (empty), 'Person Responsible' (PLAQKB), 'Type' (B), and 'Component' (BC - TWB - TST - ECA). The component field also has the text 'eCATT Extended Computer Aided Test T' next to it.

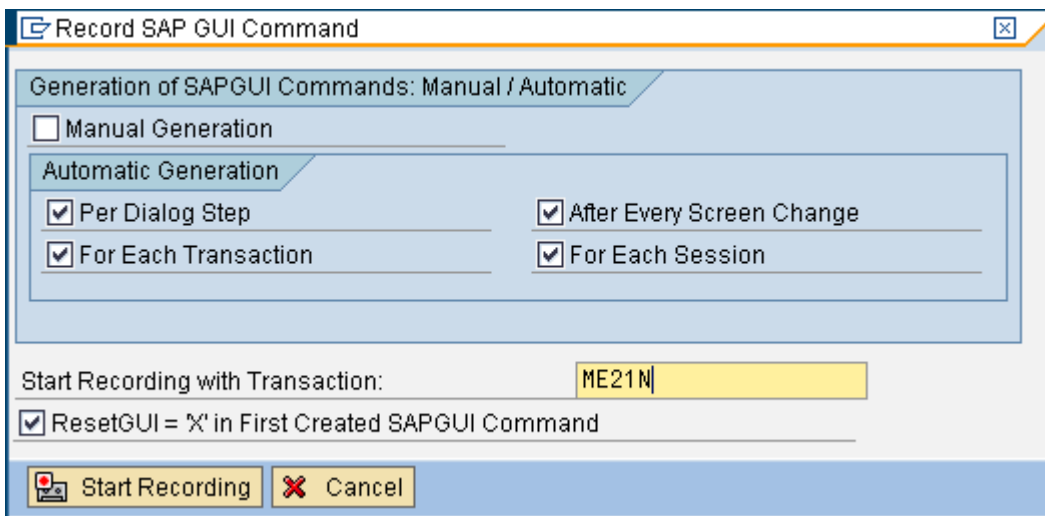
Select 'Editor' tab then select 'Pattern' button and then select Group as 'UI Control' and command as SAPGUI REC.



Enter. Below screen will appear. As this demo uses only the same system the script is created, I am choosing 'Yes'.



Below screen will appear.



The script is created based on Transaction 'ME21N', so mention that and Select 'Start Recording'.

**Record SAP GUI Command**

New Session with Following Data Was Created:

SAP System ID	ID6
Message Server	
Logon Group	
Load Balancing	
Application Server	ndainfaeas02
System Number	00
Client	800
User	PLAQKB
SAP Logon Language	EN
Session ID for Script	ses [1]
Connection ID for Script	/app/con [0]

New Sessions Are Created in Current System Data.  
Do You Want to Record This New Session?

Yes No Cancel

✓ ✗

Select 'Yes'.

**Recording Running ...**

Generation of SAPGUI Commands: Manual / Automatic

Manual Generation

Automatic Generation

Per Dialog Step  After Every Screen Change

For Each Transaction  For Each Session

Insert New SAPGUI Command

InitialState Generation Extended...

Insert GETGUI Command Insert CHEGUI Command

Close Recorded GUIs

✓ Stop Recording

The above screen will be there until all the recording is done. It will go only if 'Stop Recording' button is selected.

**Record creation of purchase order.**

Step 1:

### Create Purchase Order

Document Overview On | Hold Print Preview Messages Personal Setting

Shopping Cart NB Standard PO Vendor 1010 Doc. date 10.09.2009

Header

S	Itm	A	I	Material	Short Text	PO Quantity	O...	C	Deliv. Date	Net Price	Curr

Step 2:

### Create Purchase Order

Document Overview On | Hold Print Preview Messages Personal Setting

Shopping Cart NB Standard PO Vendor 1010 Sunny Electronics GmbH Doc. date 10.09.2009

Delivery/Invoice Conditions Texts Address Communication Partners Additional Data Org. Data

Purch. Org. 2200  
 Purch. Group 220  
 Company Code 2200

Step 3: Line Item no.10

Purch. Org. 2200 IDES France  
 Purch. Group 220 Paris  
 Company Code 2200 IDES France

S	Itm	A	I	Material	Short Text	PO Quantity	O...	C	Deliv. Date	Net Price	Curr	Per	O...	Matl Group	Plnt
				M-06		20				0.00 EUR	0				2200
										0.00 EUR	0				

Step 4:

Line Item no.20

### Create Purchase Order

Document Overview On | Hold Print Preview Messages Personal Setting

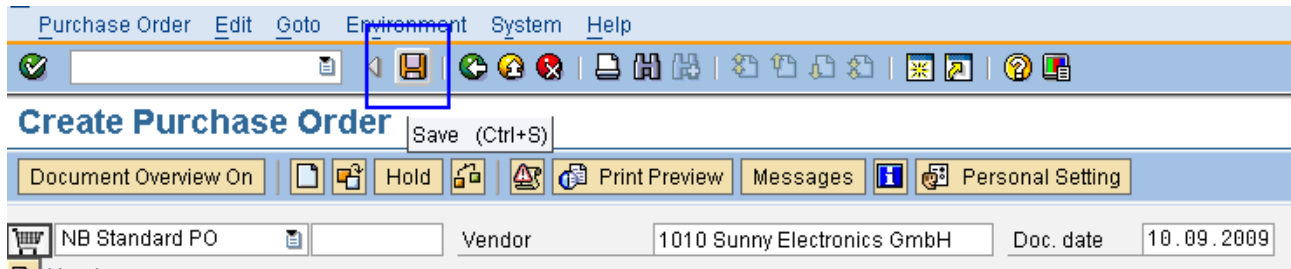
Shopping Cart NB Standard PO Vendor 1010 Sunny Electronics GmbH Doc. date 10.09.2009

Header

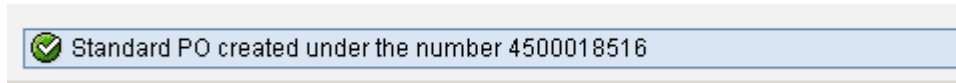
S	Itm	A	I	Material	Short Text	PO Quantity	O...	C	Deliv. Date	Net Price	Curr	Per
	10			IBM-1000	Hard Disk 40 GB		100	PC	0 10.09.2009	72.00 EUR		2
	20			IBM-1000	Hard Disk 40 GB		20	PC	0 10.09.2009	72.00 EUR		2
										0.00 EUR		0

Step 5:

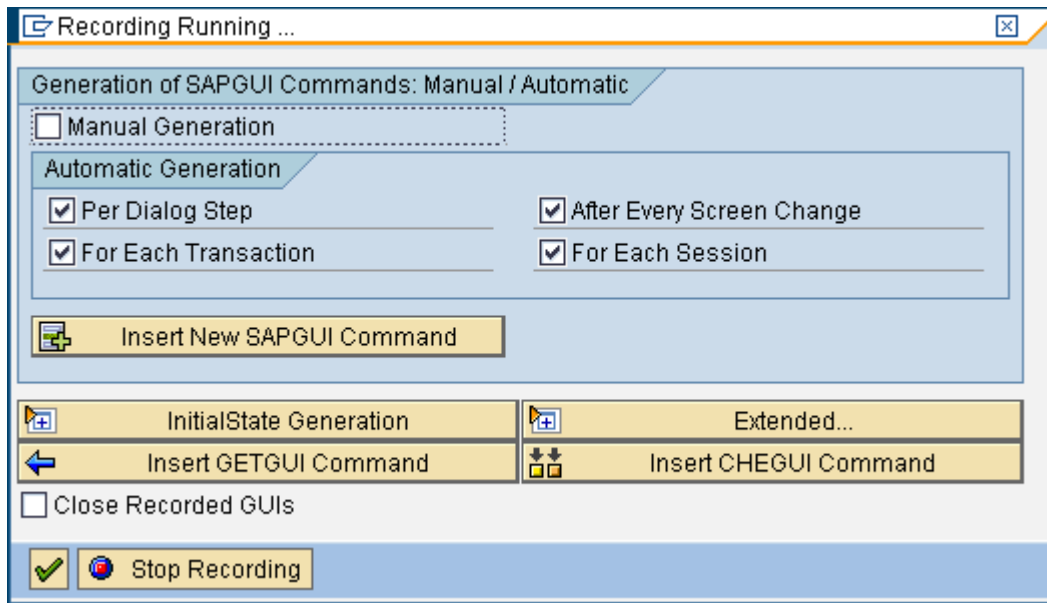
Save.



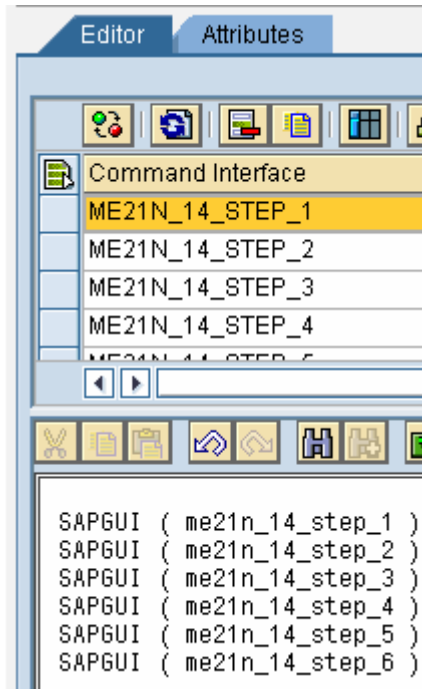
Purchase order will be created.



Select 'Stop Recording' below.



The recordings can be seen in the Editor.

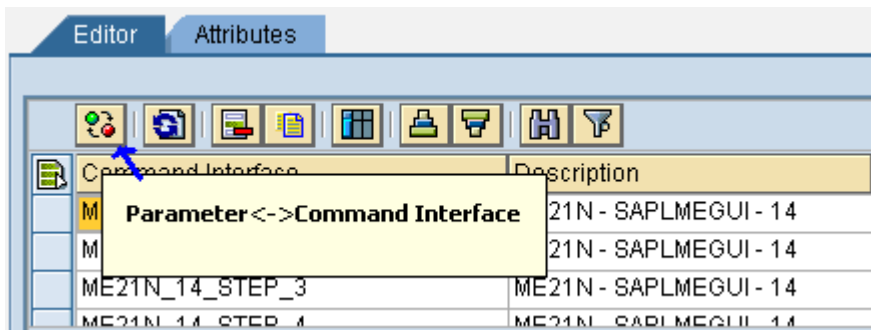


Save the script.

### Parameterization of the script

Step 1:

Select the Icon as mentioned in the screen shot below.



Step 2:

Create parameters as shown below.

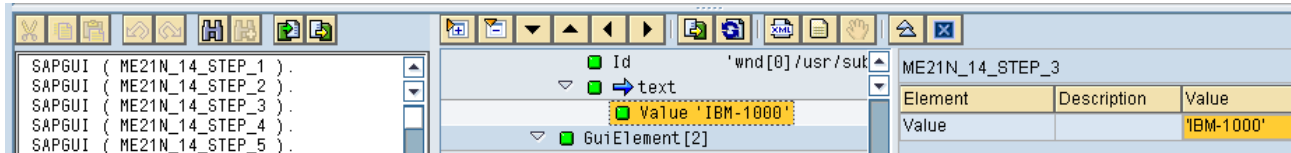
The screenshot shows the SAP Editor interface with a table. The table has five columns: 'Parameter', 'Description', 'Value', 'I/E/M', and 'Parameter Reference'. The table content is as follows:

Parameter	Description	Value	I/E/M	Parameter Reference
V_MATERIAL	Material Number		V	EMATNR
V_QUANTITY	Purchase Order Quantity		V	BSTMG
V_PLANT	Plant		V	MEPO_EWERK

Step 3:

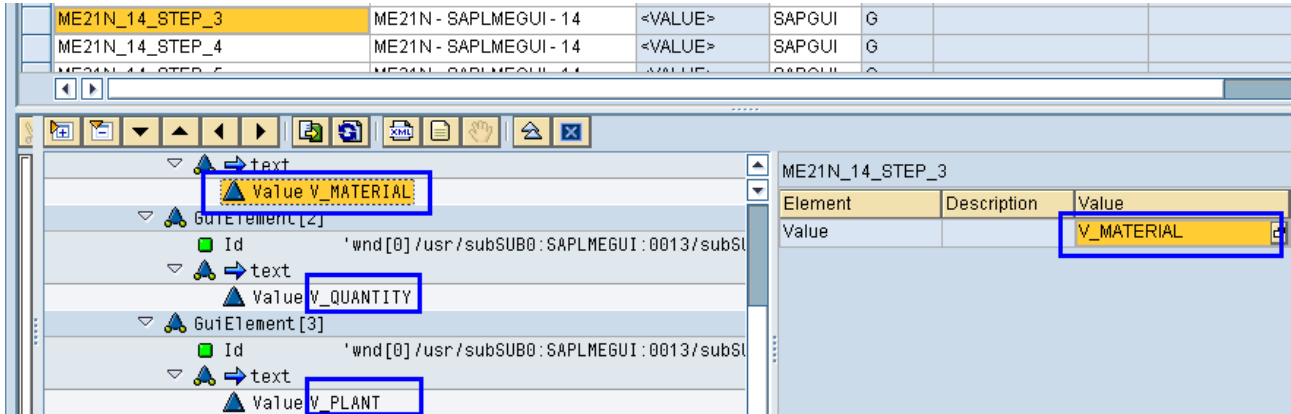
Select the recording step which contains the Material, Quantity & Plant data .Here it is ME21N\_14\_STEP\_3.





**Step 4 :**

Replace the value with parameters



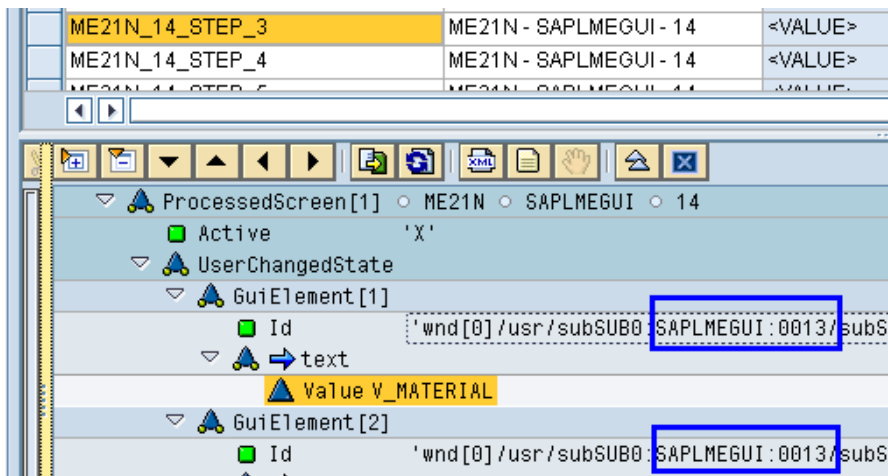
**Step 5:**

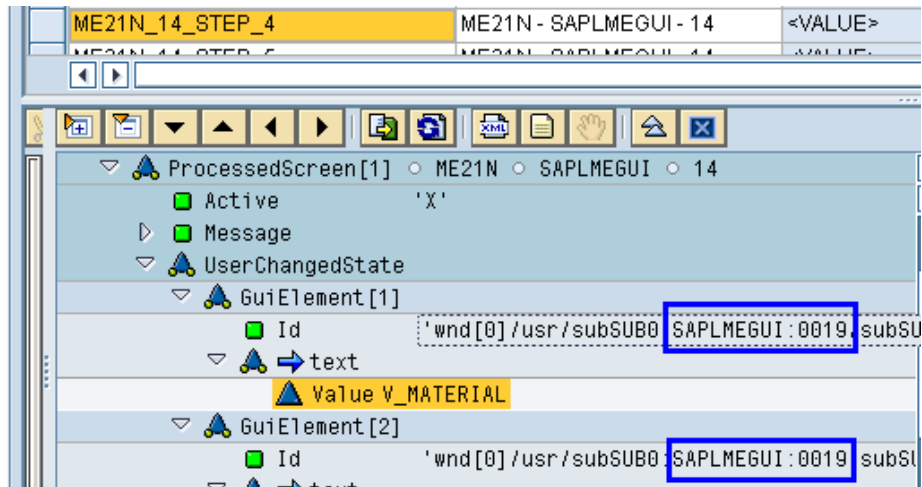
ME21N\_14\_STEP\_4 is the recording for the line item no.20 , this is also parameterized same as ME21N\_14\_STEP\_3.

It is obvious that if you raise the question, why can't we use the same recording of line item no.10?

The reason for not using the same recording is,

The value of SAPLMGUI is **0013** for ME21N\_14\_STEP\_3 whereas it is **0019** for ME21N\_14\_STEP\_4. So both are not same.





## Testing the Script:

### Pre-request for testing.

- The test file is saved in the desktop.
- The coding needs to be done in the editor to fetch the data from the desktop file and then store in an internal table.
- The rows of internal table is processed one by one inside DO...ENDO
- ME21N\_14\_STEP\_3 should be used only by the first line item. Other line items should use ME21N\_14\_STEP\_4.

### Testing the script as it is.

#### Test Data:

IBM-1000	100	2200
IBM-1000	20	2200
IBM-1000	30	2200

#### Expected Result.

When the test script is executed, Purchase order should be created with three line items as mentioned in the test data.

#### Actual result.

Once the script is executed, Purchase order is created. But there are only two line items instead of 3. As per the test data, for line item 20, the quantity should be 20. But here it is 30.

NB Standard PO		4500018536	Vendor	1010 Sunny Electronics Gr						
<table border="1"> <tr> <td>Delivery/Invoice</td> <td>Conditions</td> <td>Texts</td> <td>Address</td> <td>Communication</td> <td>Pa</td> </tr> </table>					Delivery/Invoice	Conditions	Texts	Address	Communication	Pa
Delivery/Invoice	Conditions	Texts	Address	Communication	Pa					
Purch. Org.	2200	IDES France								
Purch. Group	220	Paris								
Company Code	2200	IDES France								

S	Item	A	I	Material	Short Text	PO Quantity	O...
	10			IBM-1000	Hard Disk 40 GB	100	PC
	20			IBM-1000	Hard Disk 40 GB	30	PC

That means the line item 20 is over written by line item 30. This is the issue we will face while populating the Table Controls.

### Modification of the recording of Test Script.

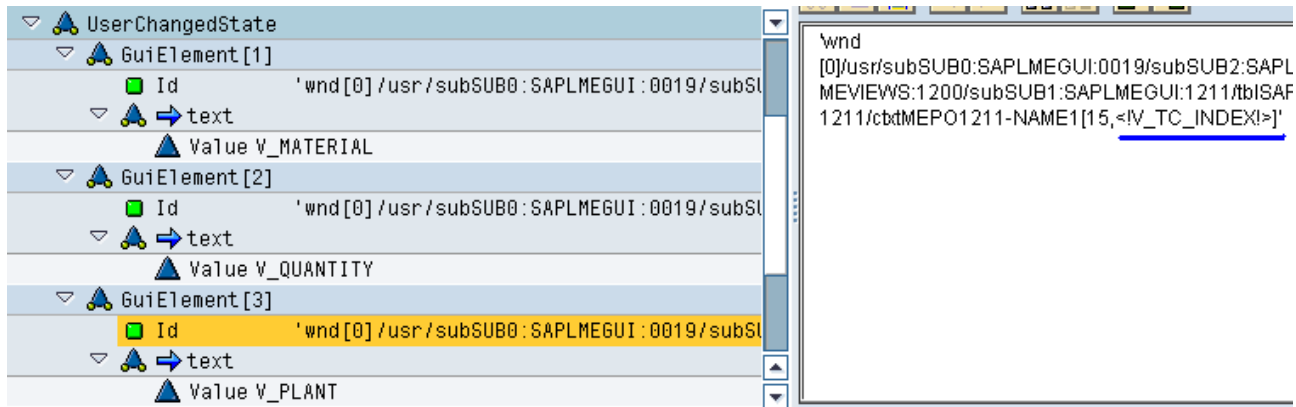
In order to make the script address consecutive rows of the table, we will need to parameterize the control ID.

Parameterize the control ID of Material in the recording ME21N\_14\_STEP\_4.

The pair of coordinates at the end ( EMATN[4,1] ) represents the cell of the table control column, row. To parameterize the row number, use a normal variable but enclose it in the escape sequence <!...!> like this:

'wnd[0] /usr /subSUB0 : SAPLMEGUI : 0019 /subSUB2 : SAPLMEVIEWS : 1100 /subSUB2 : SAPLMEVIEWS : 1200 /subSUB1 : SAPLMEGUI : 1211 /tblSAPLMEGUITC\_1211 /ctxtMEPO1211 -EMATN[4, <IV\_TC\_INDEX!>]'

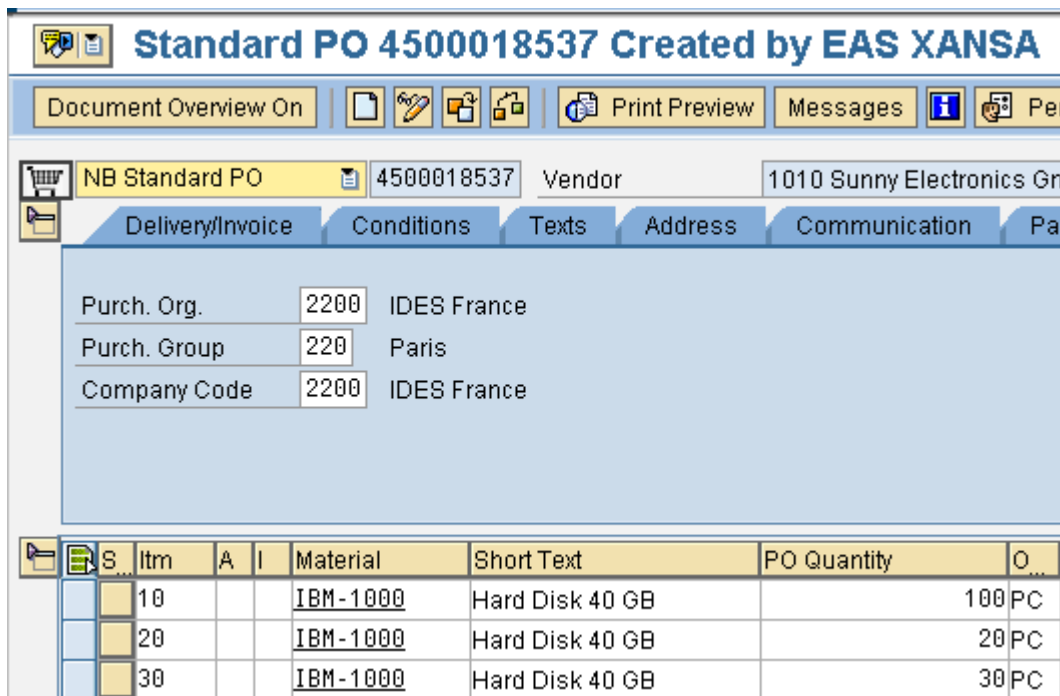
Similarly, do it for Quantity and plant.



Here the variable V\_TC\_INDEX value is '0' for first line item and then it is incremented by 1 for consecutive rows. This is handled in Editor.

### Testing after modification of Script.

Once the script is executed, Purchase order is created with three line items.



### Conclusion:

By parameterize the control ID of the parameters, the Data is successfully populated in Table controls while using eCATT.

## Related Content

[eCATT - An Introduction Blog](#)

[eCATT - Introductory Articles](#)

[Populating data into table controls in eCATT](#)

For more information, visit the [Business Intelligence 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.