

Web Dynpro ABAP- Database Manipulations using Table



Applies to:

SAP ECC 6.0. For more information, visit the [Web Dynpro ABAP homepage](#).

Summary

This article is designed for those who require support for performing Database Manipulations using Table in Web Dynpro ABAP.

Author: J.Jayanthi

Company: Siemens Information Processing Services Pvt. Ltd.

Created on: 11.09.2010

Author Bio

J.Jayanthi is a ABAP Certified professional with HR ABAP Knowledge.

Table of Contents

Prerequisite.....	3
Creating Web Dynpro	3
Component Controller.....	3
View.....	4
Web Dynpro Application	11
Code.....	11
Output.....	15
Related Content.....	17
Disclaimer and Liability Notice.....	18

Prerequisite

Before creating a table, it is necessary to know about Component, View and Window.

Component

The component is the central, reusable unit of the application project. You can create any number of views in a component and arrange them in any number of windows.

View

The view is the smallest unit of a Web Dynpro application visible for the user. The layout elements and dialog elements - for example, tables, text fields, or buttons - required for the application are arranged in a view. The view contains a controller and a controller context in which the application data to be processed is stored in a hierarchical structure. This allows the linking of the graphical elements with the application data.

Window

A window is used to group multiple views and to specify the navigation between the views. A view can only be displayed by the browser if the view is embedded in a window.

Database table ZZZ_EMP

Transp. Table	ZZZ_EMP	Active					
Short Description	Employee Details						
<div style="display: flex; justify-content: space-between;"> Attributes Delivery and Maintenance Fields Entry help/check Currency/Quantity Fields </div>							
<div style="display: flex; justify-content: space-between;"> Srch Help Predefined Type </div>							
Field	Key	Initi...	Data element	Data Ty...	Length	Decim...	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0	Client
PERNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	PERSNO	NUMC	8	0	Personnel number
ENAME	<input type="checkbox"/>	<input type="checkbox"/>	VORNA	CHAR	25	0	First Name
WAGE	<input type="checkbox"/>	<input type="checkbox"/>	ANSAL 15	CURR	15	2	Annual salary

Creating Web Dynpro

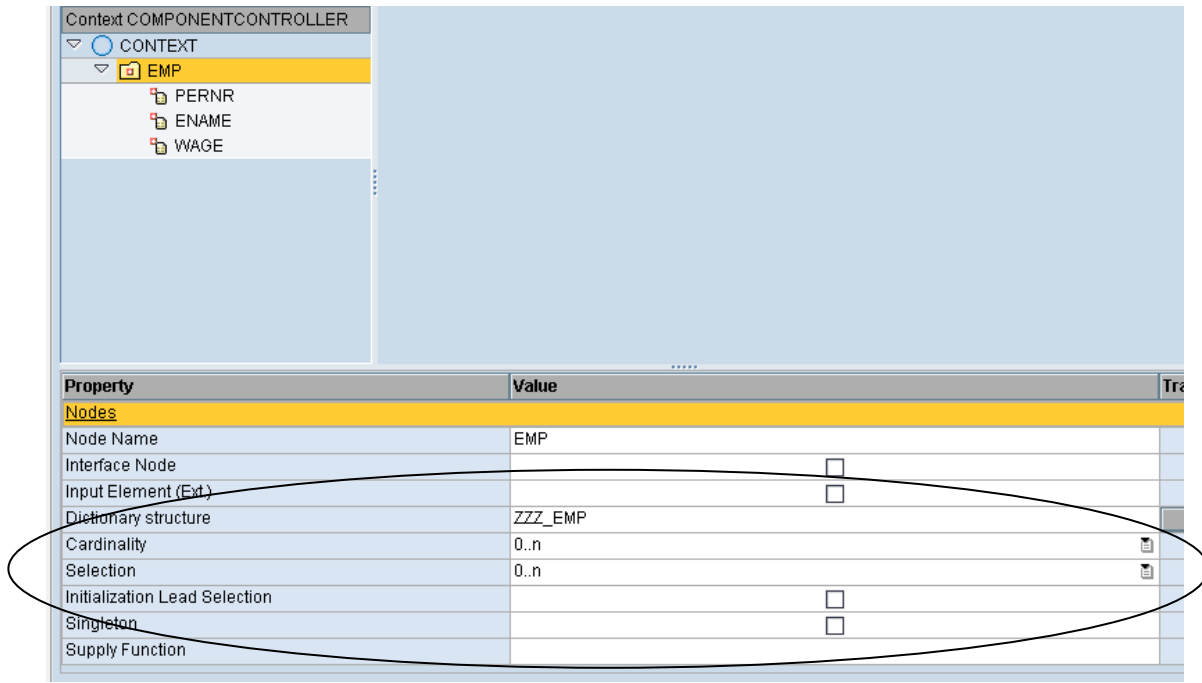
Go to SE80 and select Web Dynpro Comp./Intf. and provide the name(say ZZZ_JAYTEST14) to create. Then enter the description and choose the type as Web Dynpro Component.

Component Controller

Double click Component Controller and in the right side, right click Context->Create Node and specify database as ZZZ_EMP(mentioned in Prerequisite) and choose all the attribute except Mandt using ADD Attribute from Structure button.

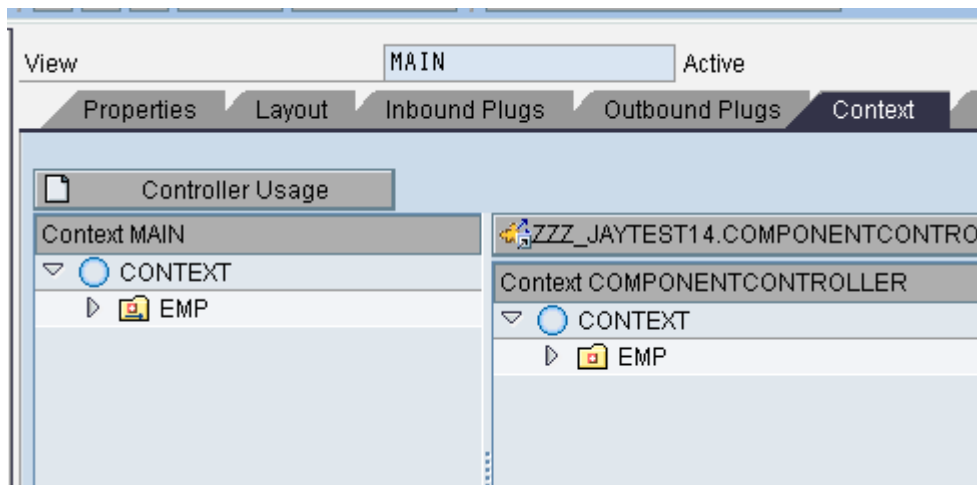
Then

1. change the cardinality to 0..n
2. change the selection to 0..n so that we can select multiple rows in table.
3. uncheck Initialization Lead Selection



View

Double click the view (Main). Go to Context and then drag EMP node from component controller context to left side context in Main.



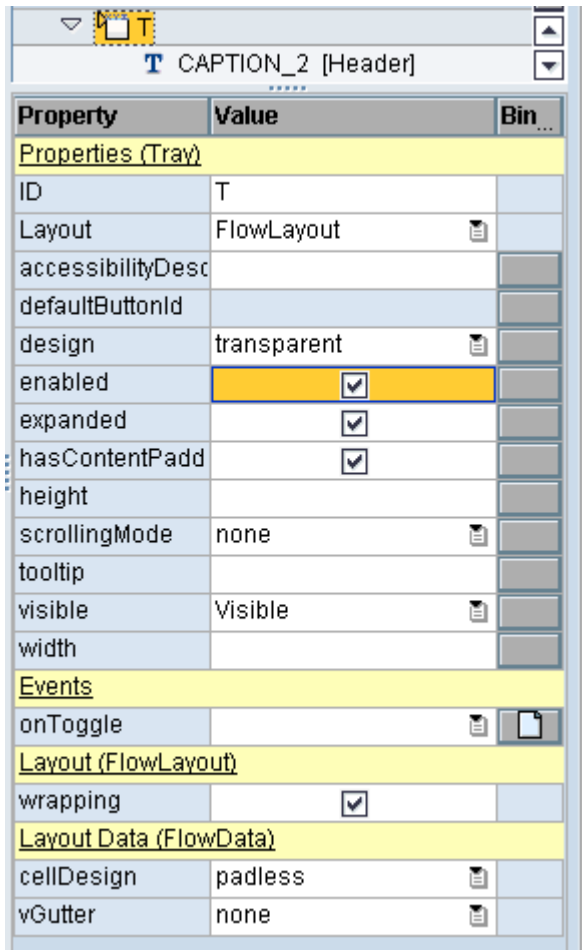
Go to Layout.

We are going to

1. Create a tray and embed a table inside it.
2. Create another tray and create four buttons inside it for Newline, Save, Delete and Refresh.

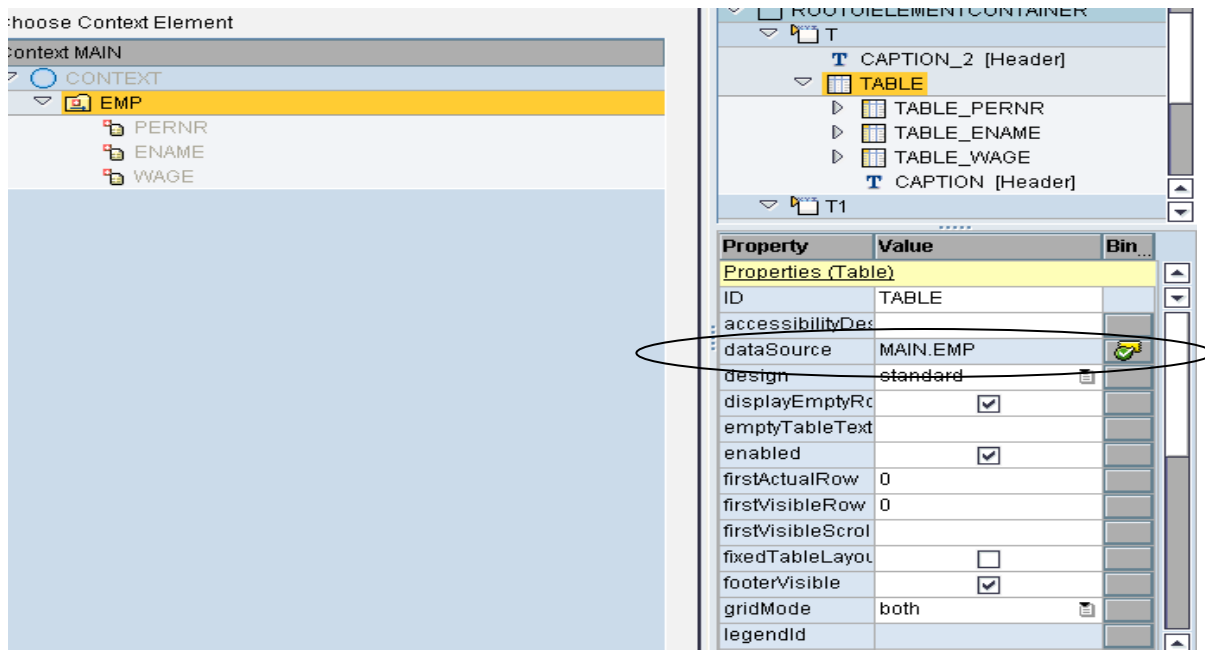
Right click ROOTUIELEMENTCONTAINER and then choose Insert Element.

Create a tray with ID say T and Caption as Database Operations.

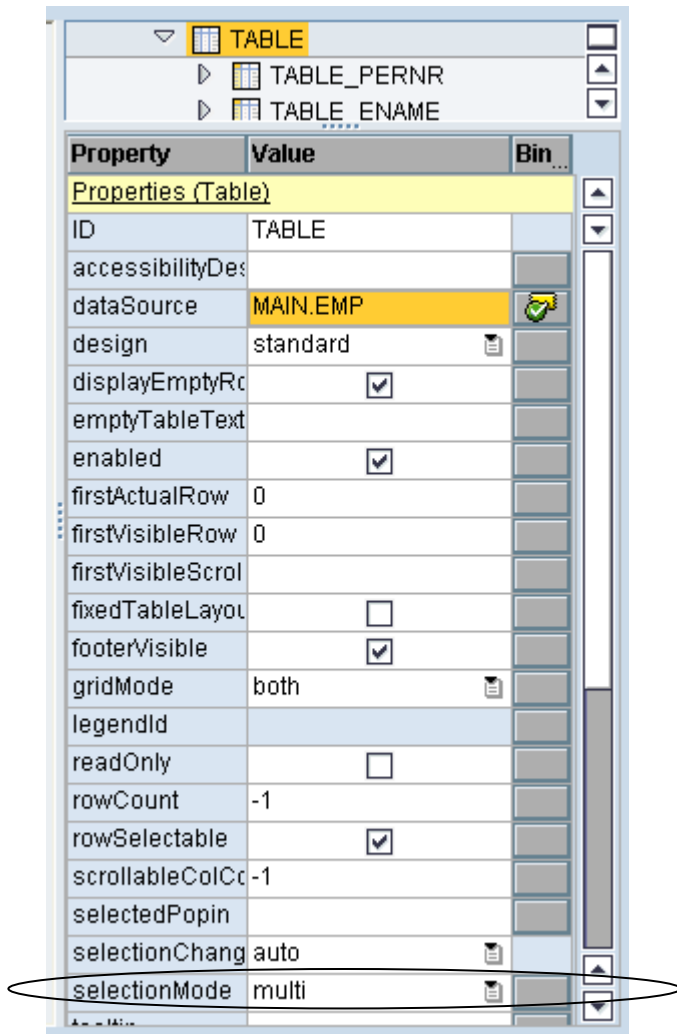


Then inside Tray, create a table with ID table using insert element.

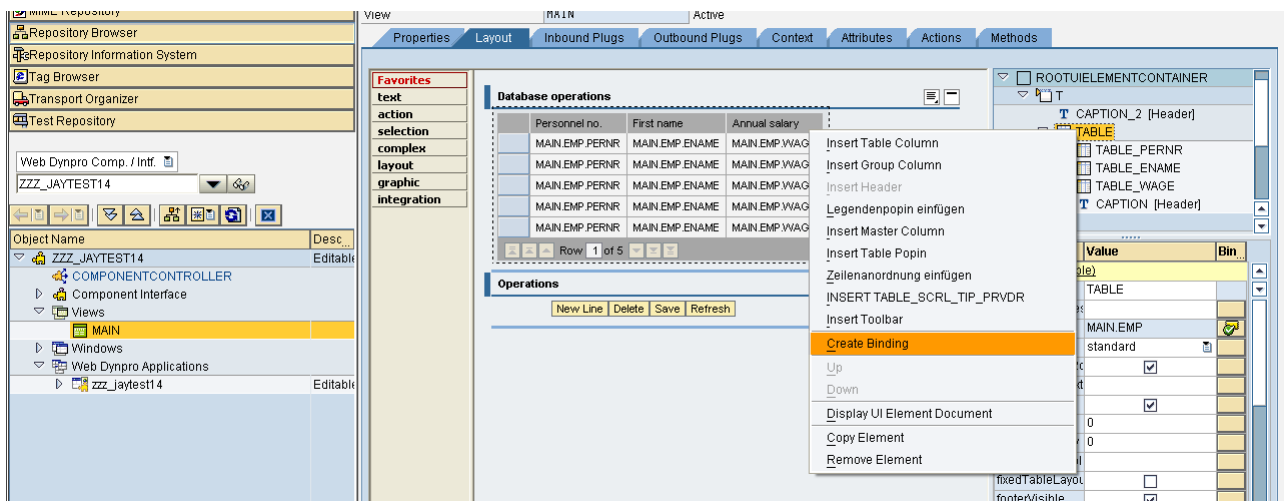
Right click the yellow color icon against Datasource in Table's properties and choose EMP as shown below.

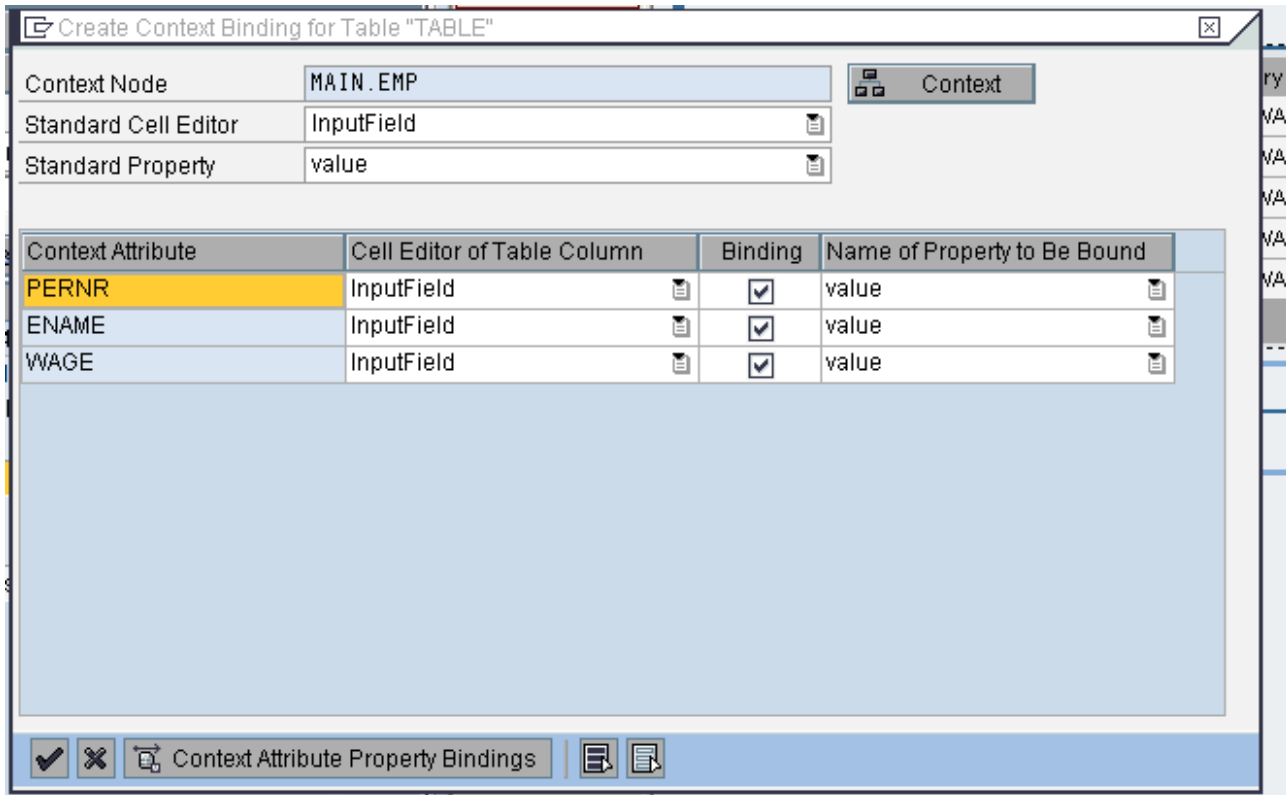


Make sure the selection mode of Table is Multi. For this selection of node needs to be 0..n.

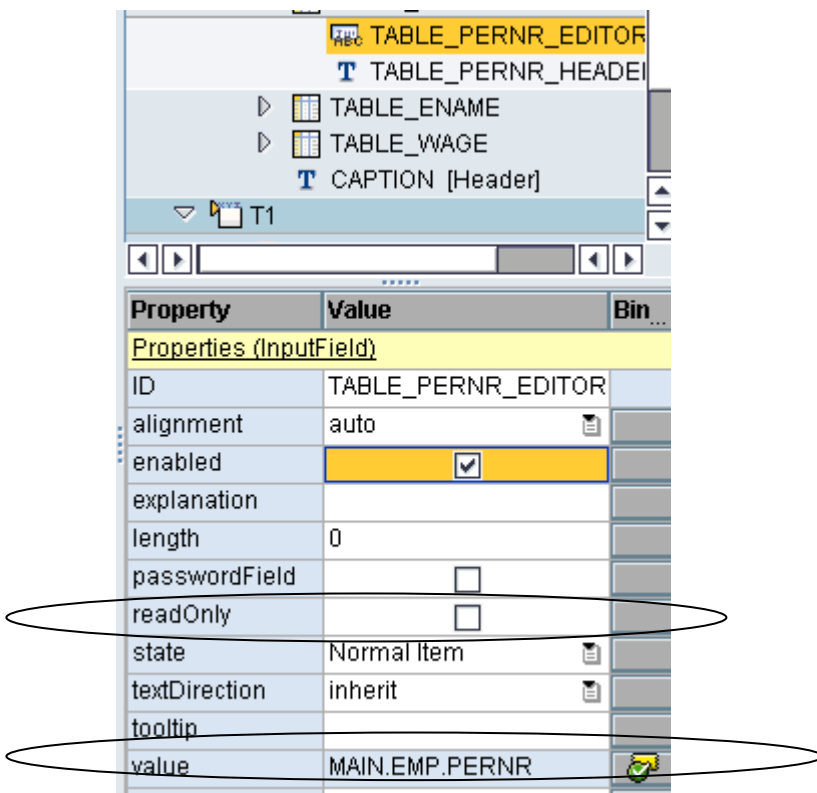


Then right click table and create Binding.





Change the value in the property of each cell editor appropriately. Similarly do for other fields ENAME and Wage.



Make sure the Readonly is not checked for cell editor inside table fields. Otherwise, during insert we cannot provide primary key values.

Design another Tray T1 with caption Operations.

Inside Tray T1, Insert Element Transparent Container (TC) and set the width of that to something so that all the buttons which we are going to design will appear in center.

Property	Value	Bin...
Properties (TransparentContainer)		
ID	TC	
Layout	FlowLayout	
accessibilityDes		
defaultButtonId		
enabled	<input checked="" type="checkbox"/>	
height		
isLayoutContain	<input checked="" type="checkbox"/>	
scrollingMode	none	
tooltip		
visible	Visible	
width	50px	

Now design four buttons by insert element with ID ADD, DELETE, UPDATE and REFRESH.

Create action for each button using create icon.

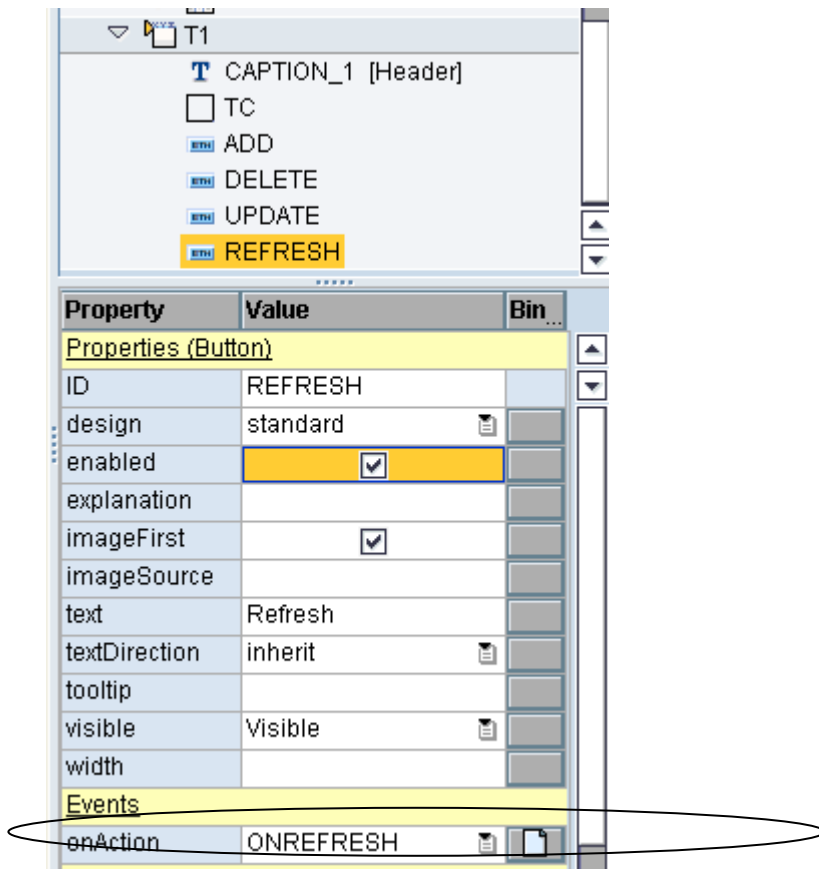
Property	Value	Bin...
Properties (Button)		
ID	ADD	
design	standard	
enabled	<input checked="" type="checkbox"/>	
explanation		
imageFirst	<input checked="" type="checkbox"/>	
imageSource		
text	New Line	
textDirection	inherit	
tooltip		
visible	Visible	
width		
Events		
onAction	ONADD	

T1
 CAPTION_1 [Header]
 TC

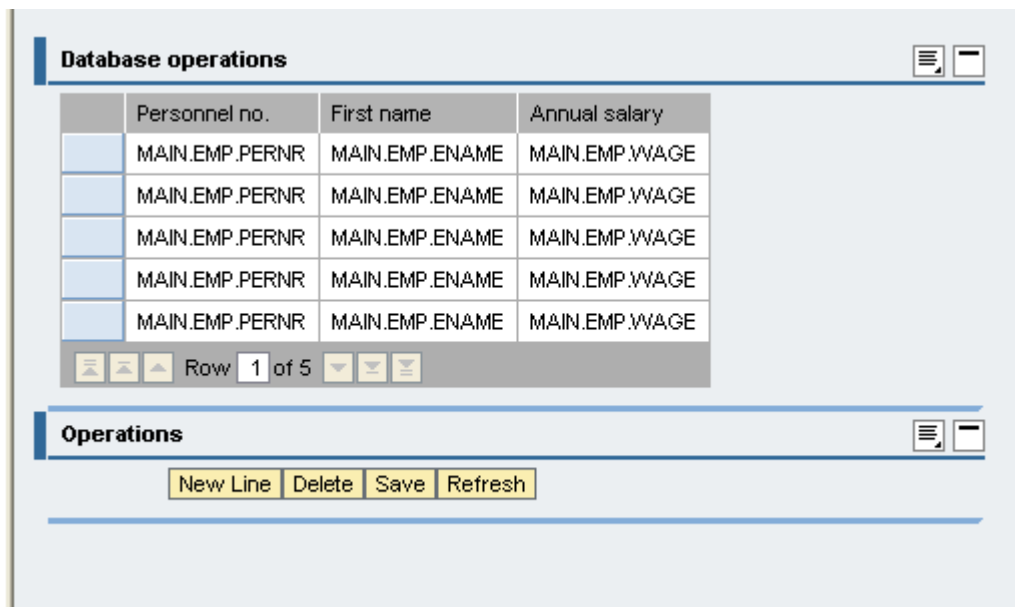
Property	Value	Bin...
Properties (Button)		
ID	DELETE	
design	standard	
enabled	<input checked="" type="checkbox"/>	
explanation		
imageFirst	<input checked="" type="checkbox"/>	
imageSource		
text	Delete	
textDirection	inherit	
tooltip		
visible	Visible	
width		
Events		
onAction	ONDELETE	
Layout Data (FlowData)		
cellDesign	padless	

T1
 CAPTION_1 [Header]
 TC

Property	Value	Bin...
Properties (Button)		
ID	UPDATE	
design	standard	
enabled	<input checked="" type="checkbox"/>	
explanation		
imageFirst	<input checked="" type="checkbox"/>	
imageSource		
text	Save	
textDirection	inherit	
tooltip		
visible	Visible	
width		
Events		
onAction	ONUPDATE	



The layout will appear as below.



Web Dynpro Application

Create Web Dynpro Application by right clicking the object (ZZZ_JAYTEST14).

Code



Use Code wizard to generate code. Our aim is to initially display all the database contents of the custom table ZZZ_EMP. So we are first going to code in WDDOINIT METHOD.

To read the context node EMP, do as below.

<input checked="" type="radio"/> Read Context	
Node/Attribute	EMP

Keep the below code and remove the rest which is not required.

Then write the selection logic using simple select statement.

Then Bind the table.

```
method WDDOINIT .
* Read Context EMP
DATA lo_nd_emp TYPE REF TO if_wd_context_node.
* navigate from <CONTEXT> to <EMP> via lead selection
lo_nd_emp = wd_context->get_child_node( name = wd_this->wdctx_emp ).
* Selection Logic
data itab type standard table of zzz_emp.
select * from zzz_emp into table itab.
* Bind Table
lo_nd_emp->bind_table( itab ).
endmethod.
```

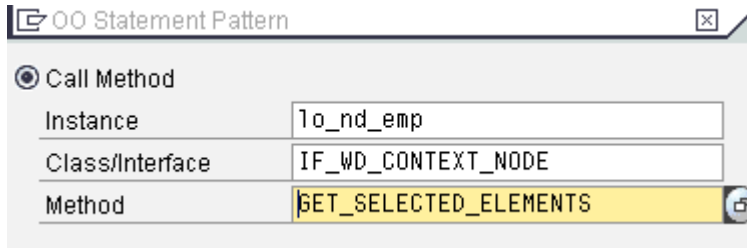
Next is how to create blank lines. This logic needs to be written in New Line Event(ONADD).

As usual, read the context node EMP and then use Bind structure as follows to create blank line.

```
method ONACTIONONADD .
* Read Context node EMP
DATA lo_nd_emp TYPE REF TO if_wd_context_node.
DATA ls_emp TYPE wd_this->element_emp.
* navigate from <CONTEXT> to <EMP> via lead selection
lo_nd_emp = wd_context->get_child_node( name = wd_this->wdctx_emp ).
* Create new blank line
data lo_new_element type ref to if_wd_context_element.
lo_new_element = lo_nd_emp->bind_structure( new_item = ls_emp
set_initial_elements = abap_false
index = 1 ).
endmethod.
```

Next we are going to handle Save(ONUPDATE).

First read context node EMP as explained above. Then get the selected rows as follows.

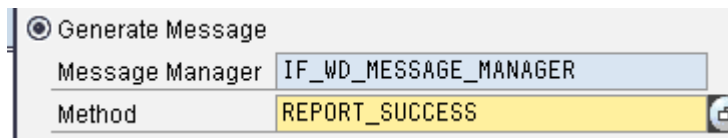


Then use Message Manager to report if there are no selected rows.



Use the normal logic to do the modifications in database for both Insert and Update.

If there is atleast one success, report the success message using Generate Message.



```

method ONACTIONONUPDATE .
  DATA lo_nd_emp TYPE REF TO if_wd_context_node.
  DATA lo_el_emp TYPE REF TO if_wd_context_element.
  DATA ls_emp TYPE wd_this->element_emp.
  DATA lt_emp TYPE wd_this->elements_emp.
  * navigate from <CONTEXT> to <EMP> via lead selection
  lo_nd_emp = wd_context->get_child_node( name = wd_this->wdctx_emp ).
  * Get the selected rows
  DATA lo_els_emp TYPE wdr_context_element_set.
  CALL METHOD lo_nd_emp->get_selected_elements
    RECEIVING
      set = lo_els_emp.
  DATA lo_api_controller TYPE REF TO if_wd_controller.
  DATA lo_message_manager TYPE REF TO if_wd_message_manager.
  IF lo_els_emp IS INITIAL.
  * get message manager
  lo_api_controller ?= wd_this->wd_get_api( ).
  CALL METHOD lo_api_controller->get_message_manager
    RECEIVING
      message_manager = lo_message_manager.
  .

```

```

* report message
CALL METHOD lo_message_manager->report_message
EXPORTING
  message_text      = 'Select Rows'
  message_type      = 0.
ELSE.
* Insert/Modify the rows in database
LOOP AT lo_els_emp INTO lo_el_emp.
  CLEAR ls_emp.
  CALL METHOD lo_el_emp->get_static_attributes
  IMPORTING
    static_attributes = ls_emp.
  APPEND ls_emp TO lt_emp.
ENDLOOP.
MODIFY zzz_emp FROM TABLE lt_emp.
IF sy-subrc EQ 0 AND sy-dbcnt GE 1.
* Populating message
lo_api_controller ?= wd_this->wd_get_api( ).
.
.
* Get message manager
CALL METHOD lo_api_controller->get_message_manager
RECEIVING
  message_manager = lo_message_manager.
CALL METHOD lo_message_manager->report_success
EXPORTING
  message_text = 'Saved Successfully'.
ELSE.
* get message manager
lo_api_controller ?= wd_this->wd_get_api( ).
CALL METHOD lo_api_controller->get_message_manager
RECEIVING
  message_manager = lo_message_manager.
* report message
CALL METHOD lo_message_manager->raise_error_message
EXPORTING
  message_text = 'Problem in Saving the values'.
ENDIF.
ENDIF.
endmethod.

```

Write the logic in the method DELETE for deleting records from database.

Read the Context Node EMP and then Get selected rows as explained above. Write the logic for deleting database entries. Use Generate Message to populate Message.

```

METHOD onactionondelete .
  DATA lo_nd_emp TYPE REF TO if_wd_context_node.
  DATA lo_e1_emp TYPE REF TO if_wd_context_element.
  DATA ls_emp TYPE wd_this->element_emp.
  DATA lt_emp TYPE wd_this->elements_emp.
  * navigate from <CONTEXT> to <EMP> via lead selection
  lo_nd_emp = wd_context->get_child_node( name = wd_this->wdctx_emp ).
  * Get the selected rows
  DATA lo_els_emp TYPE wdr_context_element_set.
  CALL METHOD lo_nd_emp->get_selected_elements
    RECEIVING
      set = lo_els_emp.
  DATA lo_api_controller TYPE REF TO if_wd_controller.
  DATA lo_message_manager TYPE REF TO if_wd_message_manager.
  IF lo_els_emp IS INITIAL.
  * get message manager
  lo_api_controller ?= wd_this->wd_get_api( ).
  CALL METHOD lo_api_controller->get_message_manager
    RECEIVING
      message_manager = lo_message_manager.
  * report message
  CALL METHOD lo_message_manager->report_message
    EXPORTING
      message_text      = 'Select Rows'
      message_type      = 0.
  ELSE.
  * Delete the rows from database
  LOOP AT lo_els_emp INTO lo_e1_emp.
    CLEAR ls_emp.
    CALL METHOD lo_e1_emp->get_static_attributes
      IMPORTING
        static_attributes = ls_emp.
    APPEND ls_emp TO lt_emp.
  ENDLOOP.
  DELETE zzz_emp FROM table lt_emp.
  IF sy-subrc eq 0 and sy-dbcnt ge 1.
  * Populating message
  lo_api_controller ?= wd_this->wd_get_api( ).
  * Get message manager
  CALL METHOD lo_api_controller->get_message_manager
    RECEIVING
      message_manager = lo_message_manager.
  CALL METHOD lo_message_manager->report_success
    EXPORTING
      message_text = 'Deleted Successfully'.
  ELSE.
  * get message manager
  lo_api_controller ?= wd_this->wd_get_api( ).
  CALL METHOD lo_api_controller->get_message_manager
    RECEIVING
      message_manager = lo_message_manager.
  * report message
  CALL METHOD lo_message_manager->report_message
    EXPORTING
      message_text      = 'Problem in Deleting'
      message_type      = 0.
  ENDIF.
  ENDIF.
ENDMETHOD.

```

Output

Database operations

Personnel no.	First name	Annual salary
00000001	A	100,00
00000002	B	200,00
00000003	C	100,00

Row 1 of 3

Operations

New Line Delete Save Refresh

Click New Line Button twice to Insert 2 lines. Then enter values and select the two new lines and modify a value in existing line as below.

Database operations

Personnel no.	First name	Annual salary
00000004	D	40,00
00000005	E	50,00
00000001	A	10,00
00000002	B	200,00
00000003	C	100,00

Row 1 of 5

Operations

New Line Delete Save Refresh

Then press Save.

Saved Successfully

Database operations

Personnel no.	First name	Annual salary
00000004	D	40,00
00000005	E	50,00
00000001	A	10,00
00000002	B	200,00
00000003	C	100,00

Row 1 of 5

Operations

New Line Delete Save Refresh

Click Refresh for refresh the contents of the table.

Select a line and press Delete.

Database operations

	Personnel no.	First name	Annual salary
<input type="checkbox"/>	00000001	A	10,00
<input type="checkbox"/>	00000002	B	200,00
<input checked="" type="checkbox"/>	00000003	C	100,00
<input type="checkbox"/>	00000004	D	40,00
<input type="checkbox"/>	00000005	E	50,00

Row 1 of 5

Operations

Deleted Successfully

Database operations

	Personnel no.	First name	Annual salary
<input type="checkbox"/>	00000001	A	10,00
<input type="checkbox"/>	00000002	B	200,00
<input checked="" type="checkbox"/>	00000003	C	100,00
<input type="checkbox"/>	00000004	D	40,00
<input type="checkbox"/>	00000005	E	50,00

Row 1 of 5

Operations

Then click Refresh.

Database operations

	Personnel no.	First name	Annual salary
<input type="checkbox"/>	00000001	A	10,00
<input type="checkbox"/>	00000002	B	200,00
<input type="checkbox"/>	00000004	D	40,00
<input type="checkbox"/>	00000005	E	50,00
<input type="checkbox"/>			

Row 1 of 4

Operations

Related Content

[Reference 1](#)

[Reference 2](#)

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.