Version 1.0
Date 13.07.2016
Status In Development
Number 5
Initial version by Igor Muntoreanu

<u>Exercise Review:</u>

In this exercise we are going to create a WebDynpro Component with the following elements:

a) Group

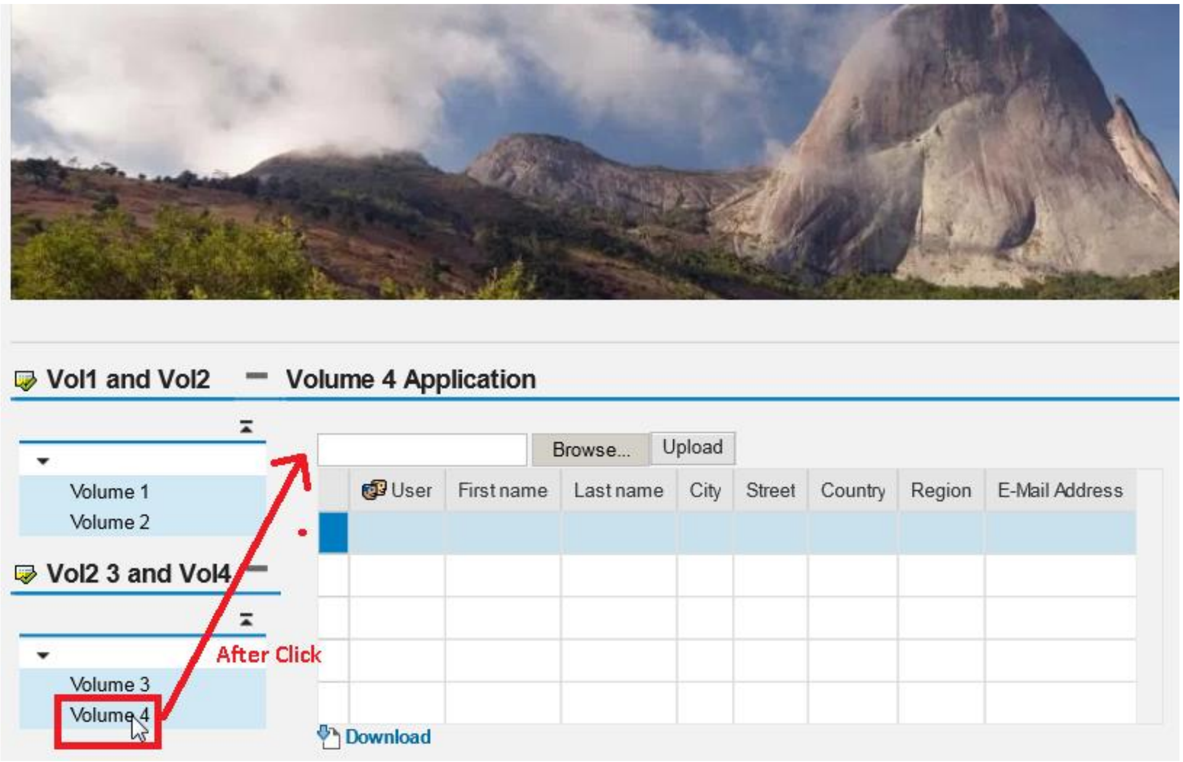
b) Tray

c) Trees

d) IFRAME

e) IMAGE

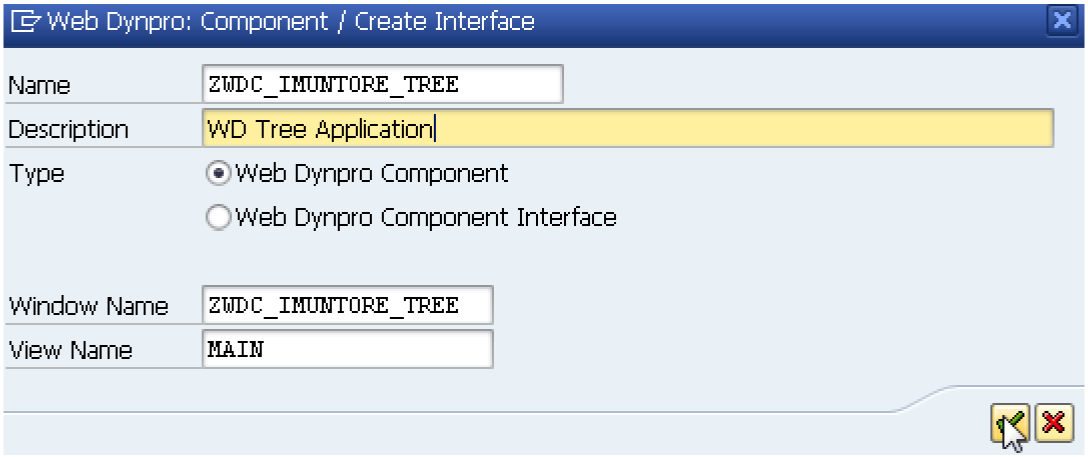
And it will have the function to load our last applications. Example:



User	First name	Last name	City	Street	Country	Region	E-Mail Address

Practical Guide:

1) Create a new WD Component:



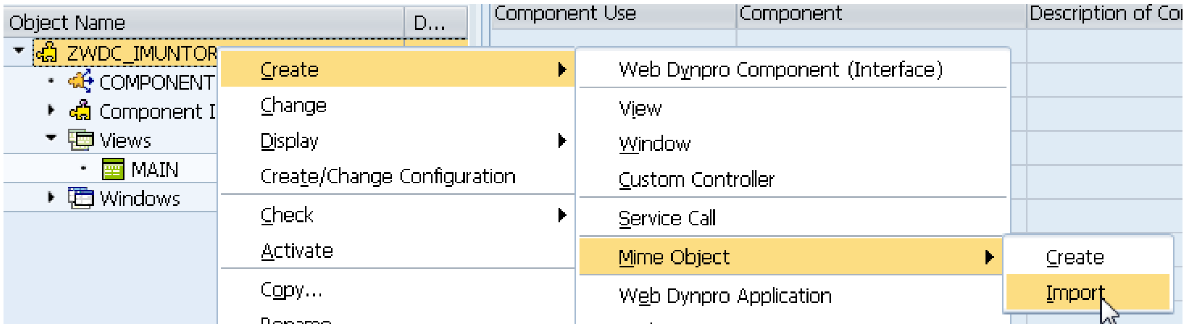
Web Dynpro: Component / Create Interface

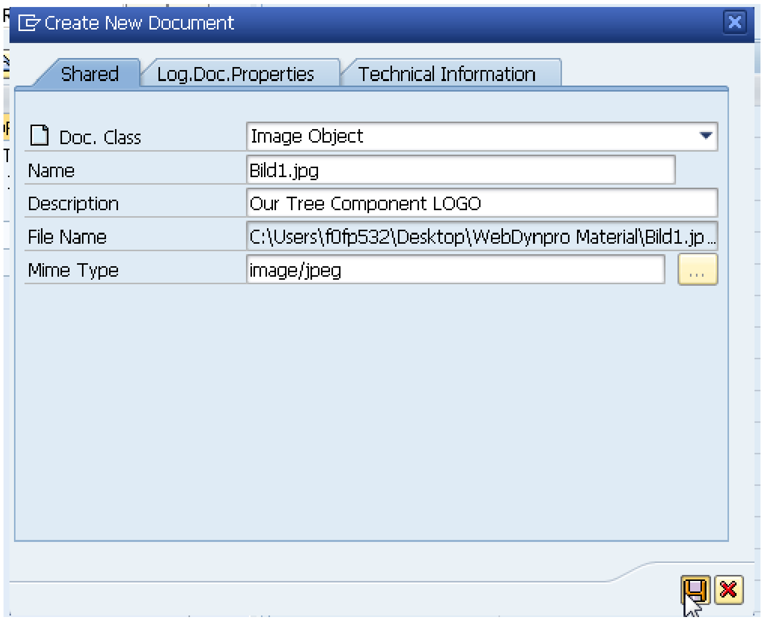
Name	ZWDC_IMUNTORE_TREE
Description	WD Tree Application
Type	● Web Dynpro Component ○ Web Dynpro Component Interface
Window Name	ZWDC_IMUNTORE_TREE
View Name	MAIN

2) Save and Activate.

3) Upload the following image as a MIME Object:

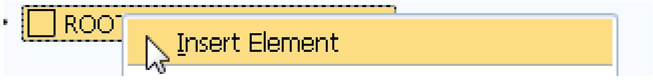
Bild1.jpg

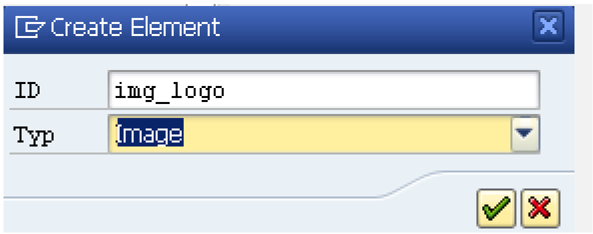




Doc. Class: Image Object
Name: Bild1.jpg
Description: Our Tree Component LOGO
File Name: C:\Users\f0fp532\Desktop\WebDynpro Material\Bild1.jp...
Mime Type: image/jpeg

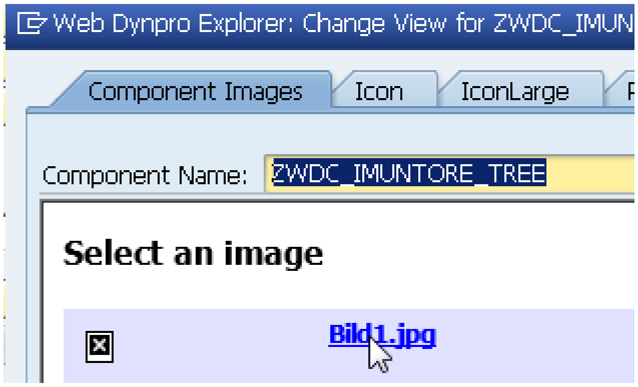
4) Go to the MAIN view
Insert the new Image:





ID: img_logo
Typ: Image

At the source property, select our uploaded image:

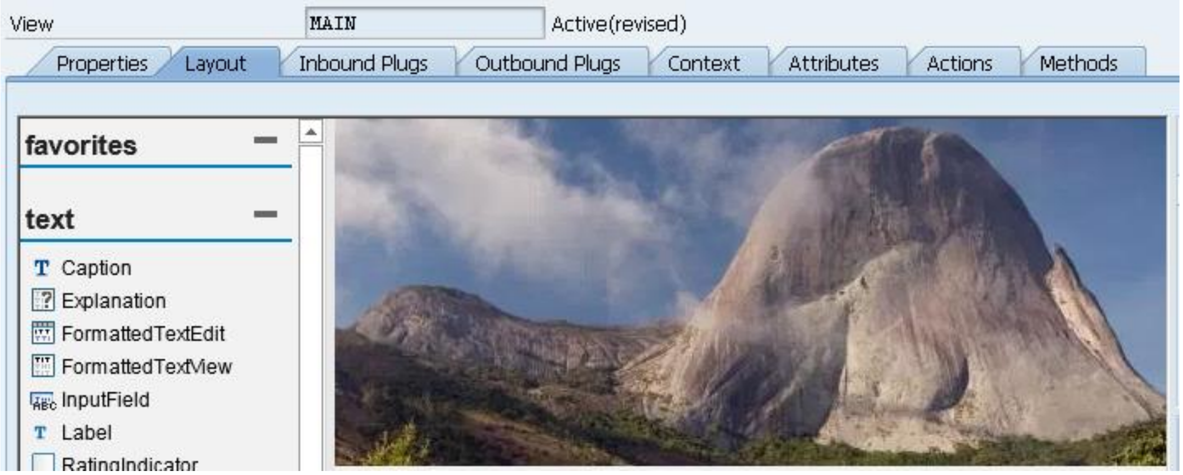


Component Name: ZWDC_IMUNTORE_TREE

Select an image

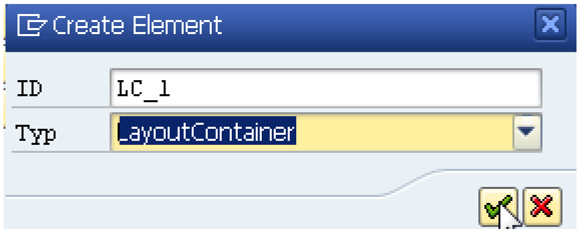
☒ Bild1.jpg

You View will look like this:



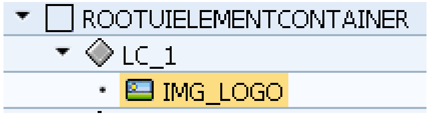
View MAIN Active(revised)

Create a Layout Container as shown below and put the IMAGE inside the new container:



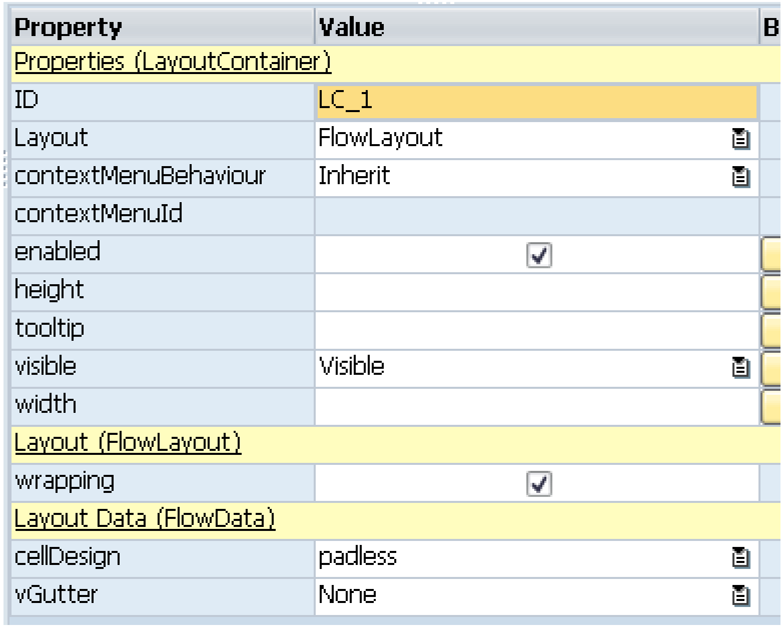
Create Element

ID	LC_1
Typ	LayoutContainer



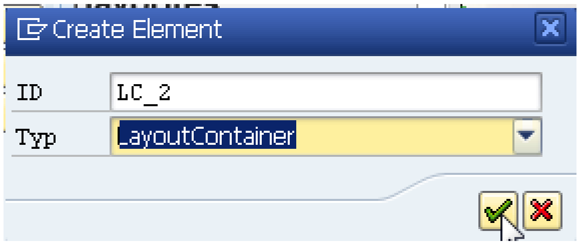
- ROOTUIELEMENTCONTAINER
  - LC_1
    - IMG_LOGO

Set the property of the layout container as demonstrated below:



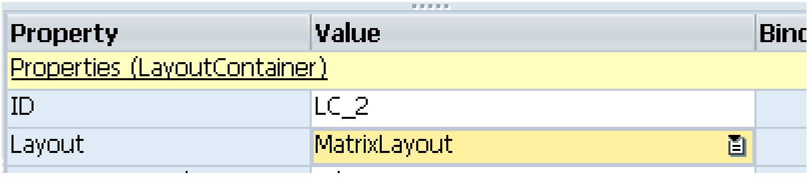
Property	Value	B
Properties (LayoutContainer)		
ID	LC_1	
Layout	FlowLayout	
contextMenuBehaviour	Inherit	
contextMenuId		
enabled	☑	
height		
tooltip		
visible	Visible	
width		
Layout (FlowLayout)		
wrapping	☑	
Layout Data (FlowData)		
cellDesign	padless	
vGutter	None	

Create another Layout Container element:



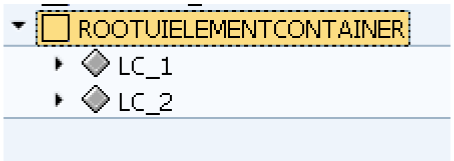
ID: LC_2
Typ: LayoutContainer

Set the Layout property as Matrix:



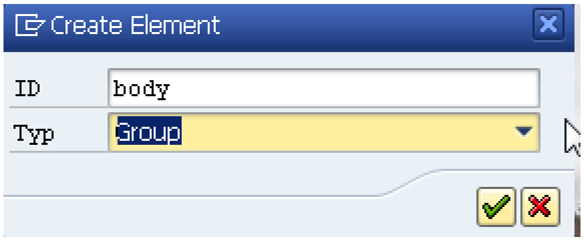
Property	Value	Binc
Properties (LayoutContainer)		
ID	LC_2	
Layout	MatrixLayout	

Result:



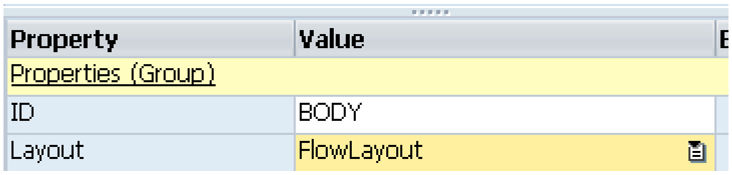
```
ROOTUIELEMENTCONTAINER
    LC_1
    LC_2
```

Create a Group Element inside the LC_2:



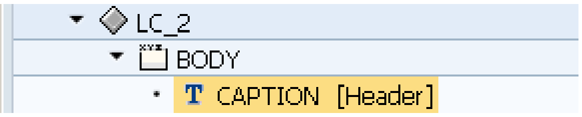
ID: body
Typ: Group

Set the layout property as FlowLayout:

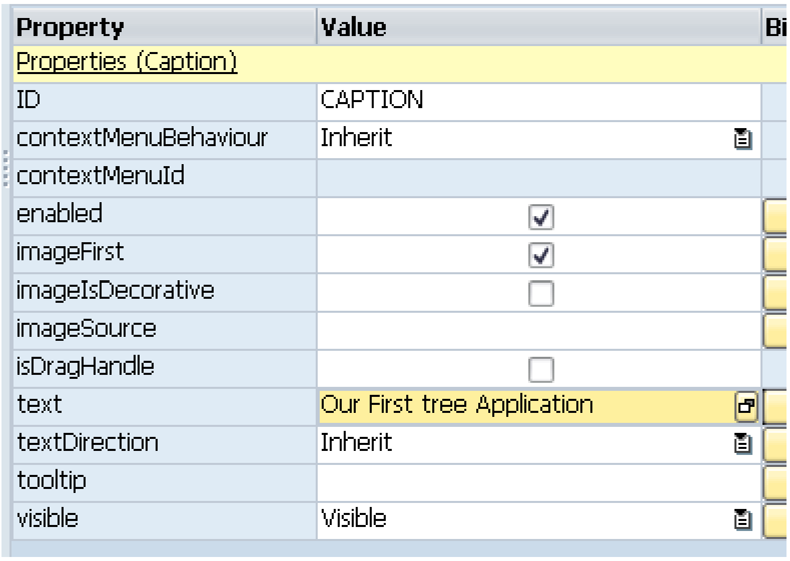


Property	Value	E
Properties (Group)		
ID	BODY	
Layout	FlowLayout	

Put some text in the automatic Label generated:

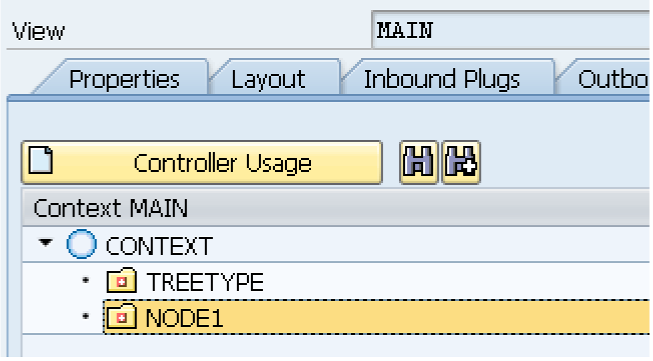


```
LC_2
    BODY
        CAPTION [Header]
```



Property	Value	Bi
Properties (Caption)		
ID	CAPTION	
contextMenuBehaviour	Inherit	
contextMenuId		
enabled	☑	
imageFirst	☑	
imageIsDecorative	☐	
imageSource		
isDragHandle	☐	
text	Our First tree Application	
textDirection	Inherit	
tooltip		
visible	Visible	

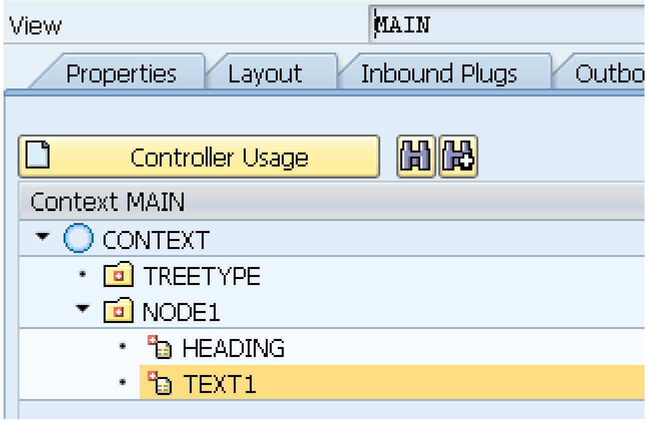
Go to the Context tab and create two nodes with the standard specifications:



View MAIN

Context MAIN
- CONTEXT
  - TREETYPE
  - NODE1

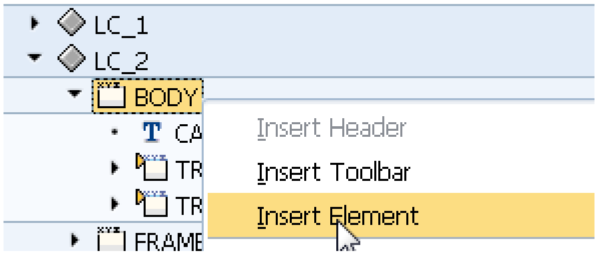
Create 2 Attributes inside NODE1,both STRING type:

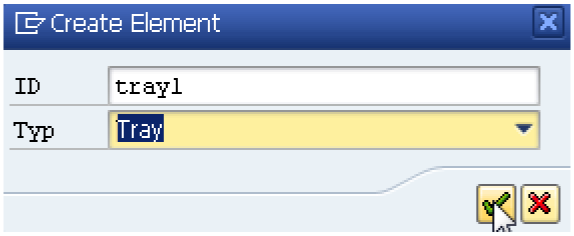


View MAIN

Context MAIN
- CONTEXT
  - TREETYPE
  - NODE1
    - HEADING
    - TEXT1

Create a Tray element inside the BODY group:

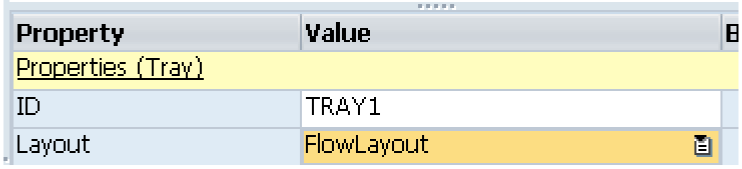




ID tray1

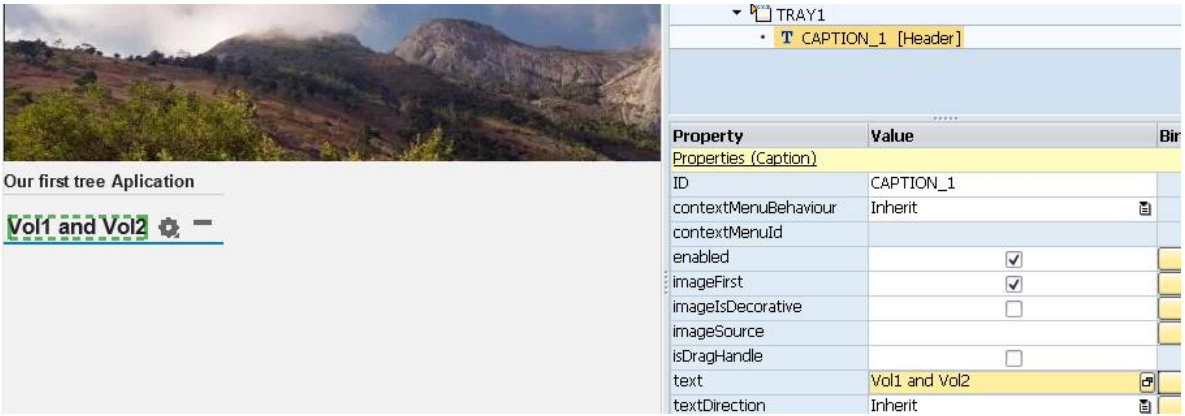
Typ Tray

Set the layout property as below:



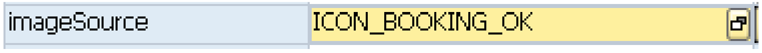
Property	Value
Properties (Tray)	
ID	TRAY1
Layout	FlowLayout

In the auto generated label, type the text:



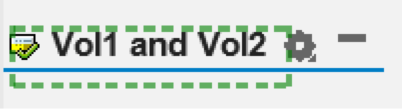
Property	Value
Properties (Caption)	
ID	CAPTION_1
contextMenuBehaviour	Inherit
contextMenuId	
enabled	☑
imageFirst	☑
imageIsDecorative	☐
imageSource	
isDragHandle	☐
text	Vol1 and Vol2
textDirection	Inherit

Set the imagesource as

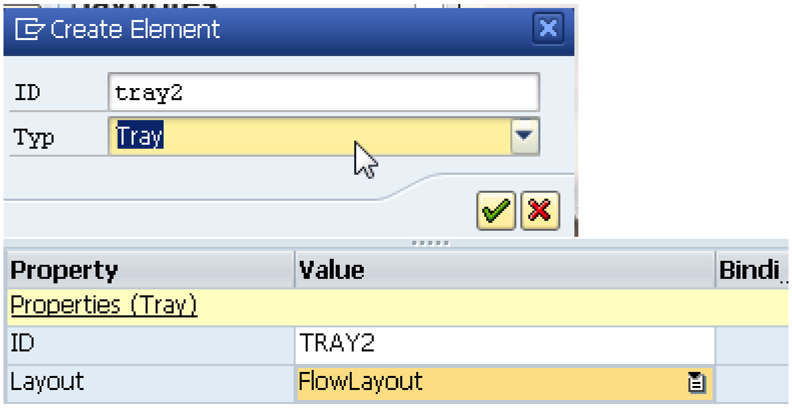


imageSource	ICON_BOOKING_OK

Result:



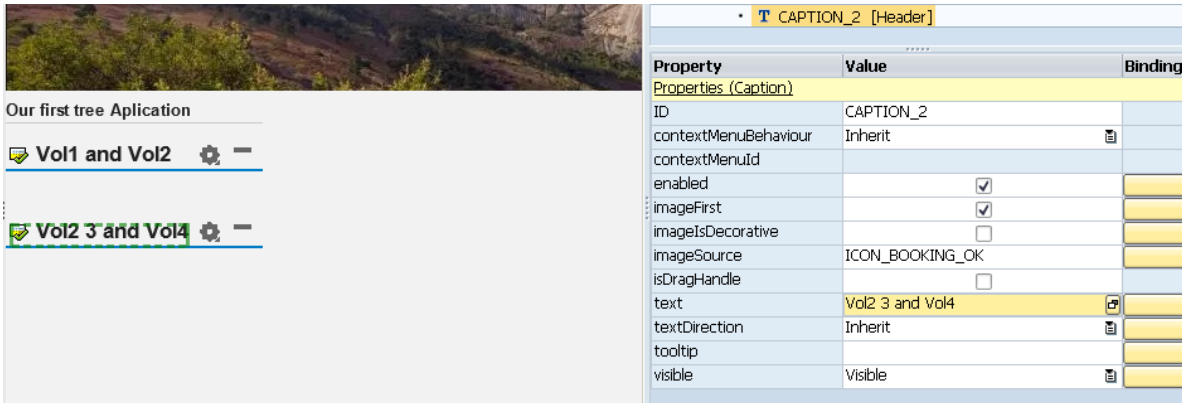
Create the TRAY2 element also inside the BODY group:



ID	tray2
Typ	Tray

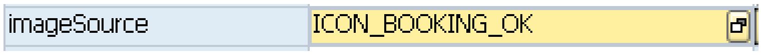
Property	Value	Bindi
Properties (Tray)		
ID	TRAY2	
Layout	FlowLayout	

In the auto generated label, type the text:



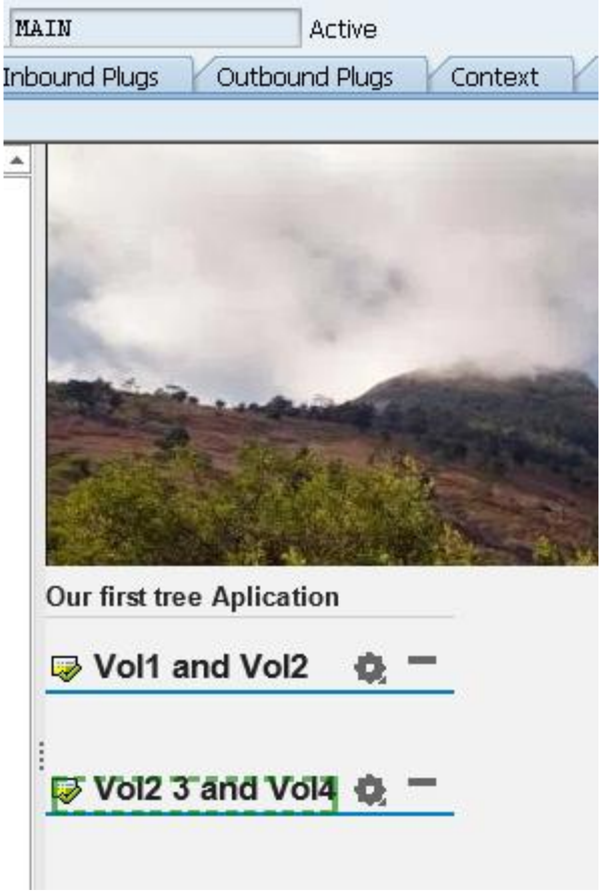
Property	Value	Binding
Properties (Caption)		
ID	CAPTION_2	
contextMenuBehaviour	Inherit	
contextMenuId		
enabled	☑	
imageFirst	☑	
imageIsDecorative	☐	
imageSource	ICON_BOOKING_OK	
isDragHandle	☐	
text	Vol2 3 and Vol4	
textDirection	Inherit	
tooltip		
visible	Visible	

Set the imagesource as

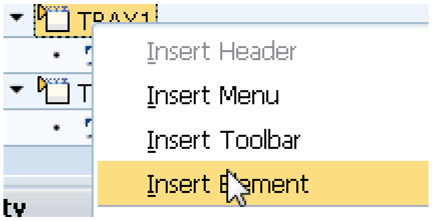


imageSource	ICON_BOOKING_OK

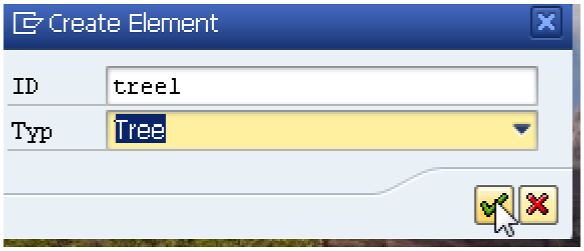
Result:



Inside the Tray1, insert the following element:



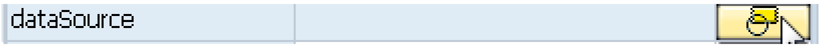
Create the Tree1 as shown below:

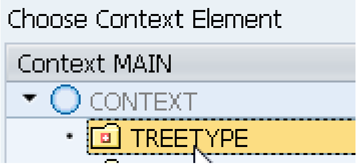


ID tree1

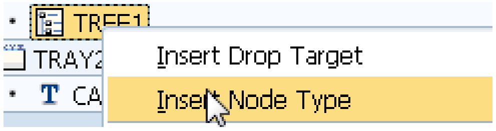
Typ Tree

Bind the dataSource:

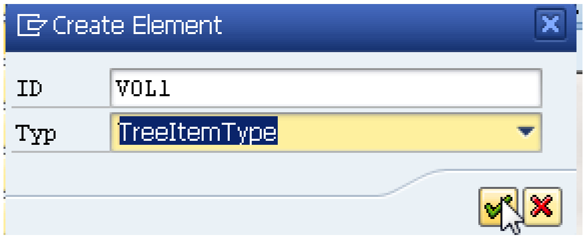




After the TREE1 created, insert the Node Type inside the TREE element.



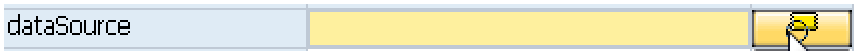
Name as VOL1, as shown below:

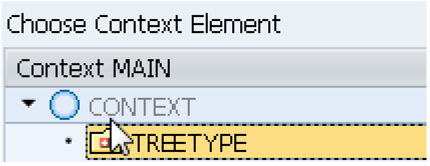


ID VOL1

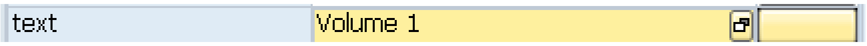
Typ TreeItemType

Bind also with the Treetype



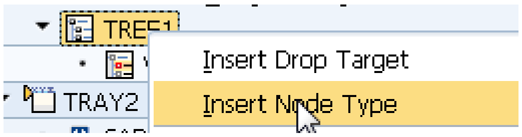


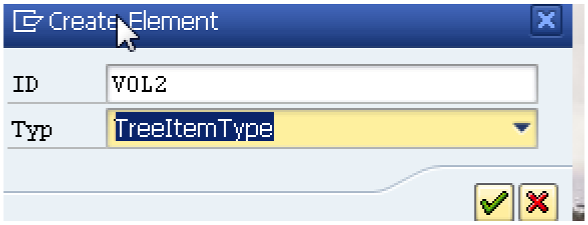
In the text property, set Volume 1.



text Volume 1

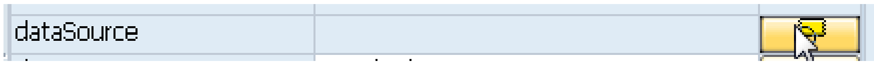
Do this step again for TreeItemType VOL2, as shown below:

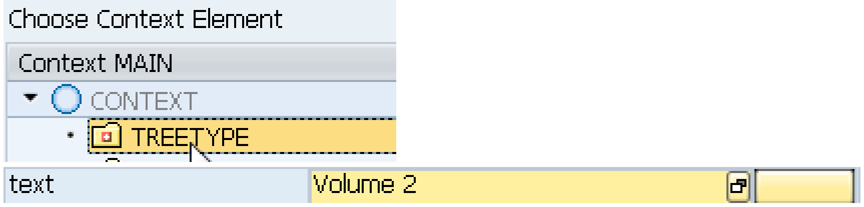




ID VOL2

Typ TreeItemType





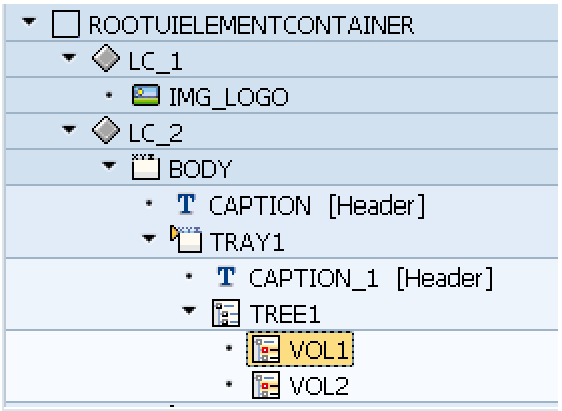
Choose Context Element

Context MAIN

```
▾ CONTEXT
    • TREETYPE
```

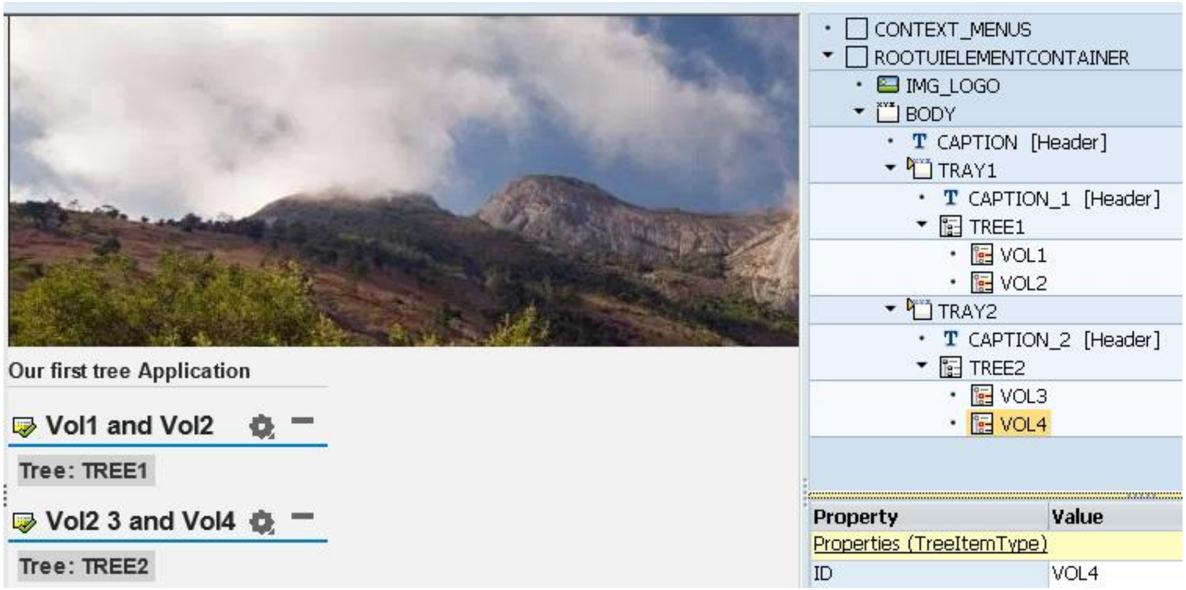
text	Volume 2

Until now, your layout must look like this:



```
▾ ROOTUIELEMENTCONTAINER
    ▾ LC_1
        • IMG_LOGO
    ▾ LC_2
        ▾ BODY
            • CAPTION  [Header]
            ▾ TRAY1
                • CAPTION_1  [Header]
                ▾ TREE1
                    • VOL1
                    • VOL2
```

Do exactly the same steps above to the TRAY2, the final result will look like this:



Our first tree Application

Vol1 and Vol2

Tree: TREE1

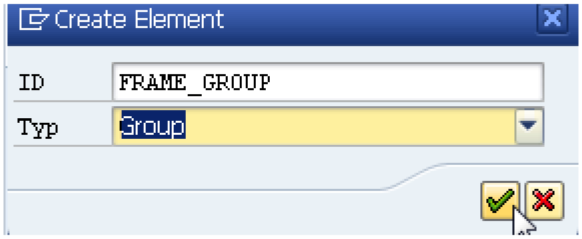
Vol2 3 and Vol4

Tree: TREE2

```
• CONTEXT_MENUS
▾ ROOTUIELEMENTCONTAINER
    • IMG_LOGO
    ▾ BODY
        • CAPTION  [Header]
        ▾ TRAY1
            • CAPTION_1  [Header]
            ▾ TREE1
                • VOL1
                • VOL2
        ▾ TRAY2
            • CAPTION_2  [Header]
            ▾ TREE2
                • VOL3
                • VOL4
```

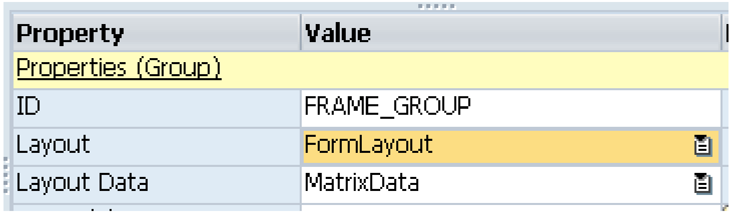
Property	Value
Properties (TreeItemType)	
ID	VOL4

Insert a new Group inside the LC_2 Layout Container:



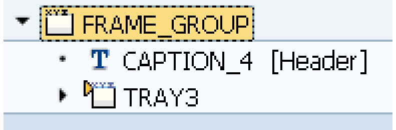
ID	FRAME_GROUP
Typ	Group

FRAME_GROUP properties:



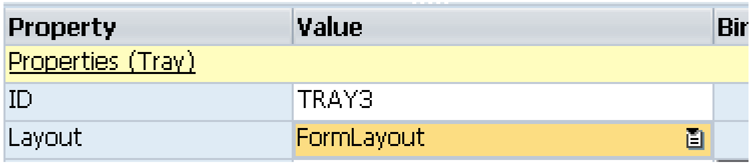
Property	Value
Properties (Group)	
ID	FRAME_GROUP
Layout	FormLayout
Layout Data	MatrixData

Indie the FRAME_GROUP, create the TRAY3:



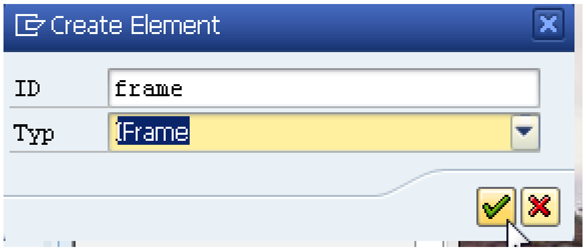
- FRAME_GROUP
  - CAPTION_4 [Header]
  - TRAY3

TRAY3 properties:



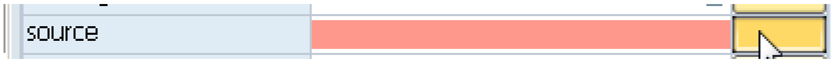
Property	Value	Bir
Properties (Tray)		
ID	TRAY3	
Layout	FormLayout	

Insert a new element inside the TRAY3:

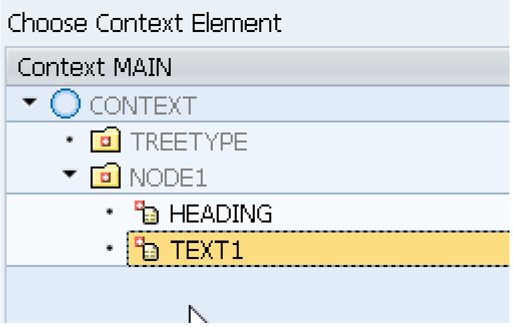


ID	frame
Typ	IFrame

Bind it with the TEXT1 attribute:



source	

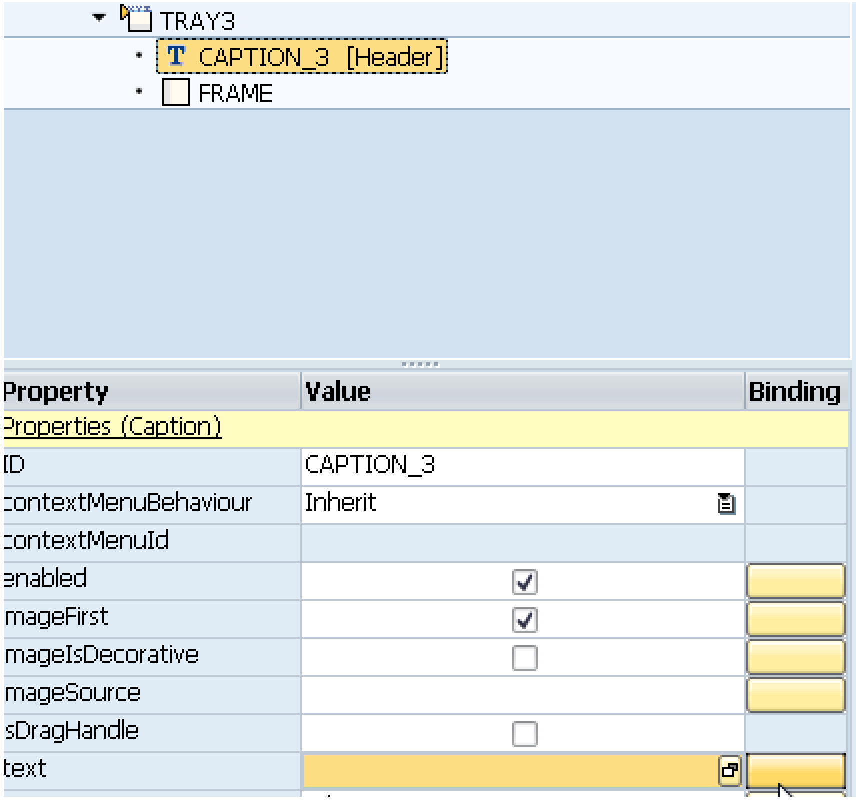


Choose Context Element

Context MAIN

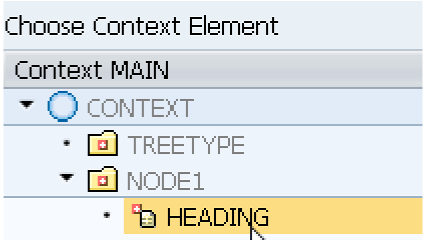
- CONTEXT
  - TREETYPE
  - NODE1
    - HEADING
    - TEXT1

In the auto generated Captionfor the Tray3, bind the text as shown below:



- TRAY3
  - CAPTION_3 [Header]
  - FRAME

Property	Value	Binding
Properties (Caption)		
ID	CAPTION_3	
contextMenuBehaviour	Inherit	
contextMenuId		
enabled	☑	
mageFirst	☑	
mageIsDecorative	□	
mageSource		
sDragHandle	□	
text		

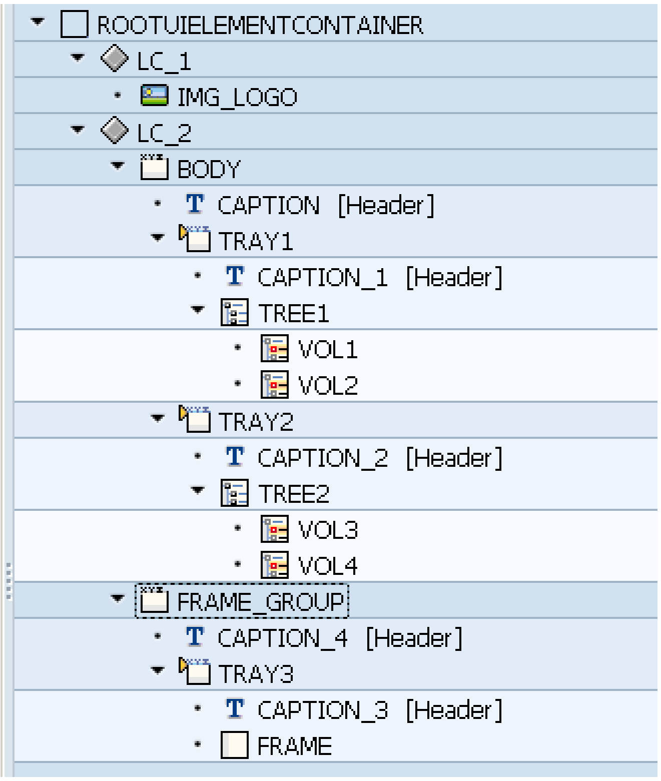


Choose Context Element

Context MAIN

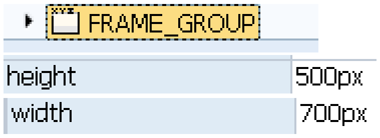
- CONTEXT
  - TREETYPE
  - NODE1
    - HEADING

Before we create the actions, let's do some layout adjustments.
First of all, your ROOT must look like this:



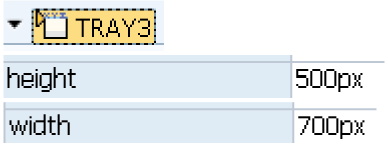
```
ROOTUIELEMENTCONTAINER
  LC_1
    IMG_LOGO
  LC_2
    BODY
      CAPTION  [Header]
      TRAY1
        CAPTION_1  [Header]
        TREE1
          VOL1
          VOL2
      TRAY2
        CAPTION_2  [Header]
        TREE2
          VOL3
          VOL4
    FRAME_GROUP
      CAPTION_4  [Header]
      TRAY3
        CAPTION_3  [Header]
        FRAME
```

Change the following properties of the following elements:



FRAME_GROUP

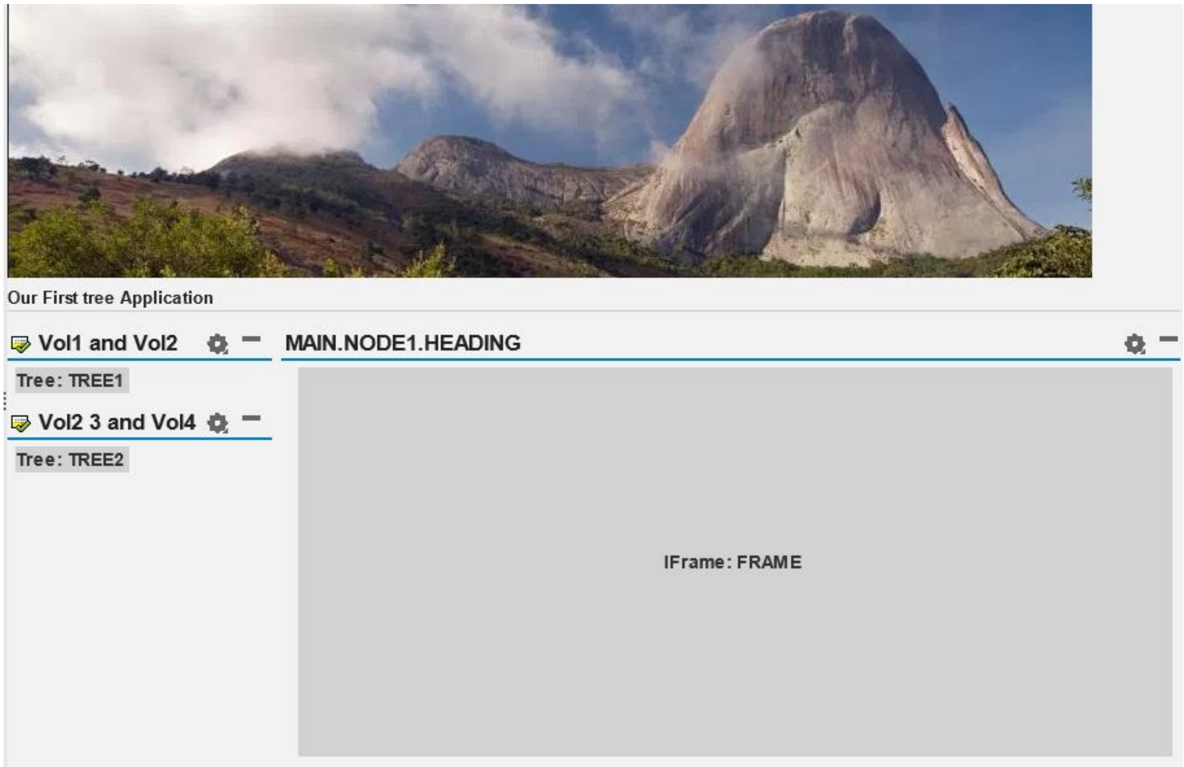
height	500px
width	700px



TRAY3

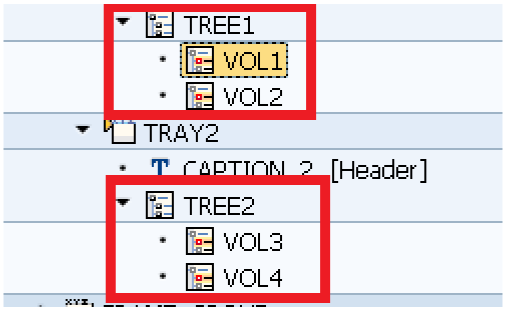
height	500px
width	700px

5) Your layout should look like this now:

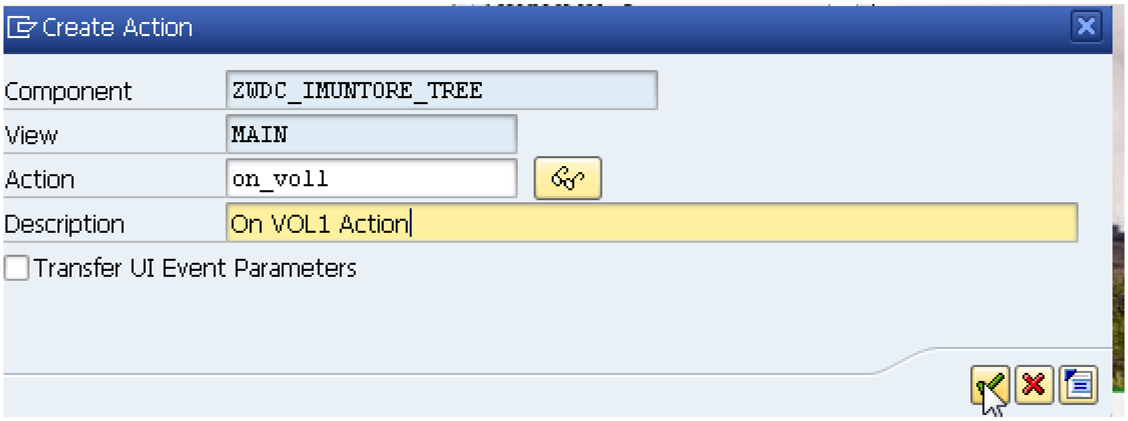


6) Creating the TreeItemTypes Actions

We will now create the actions for the following elements:



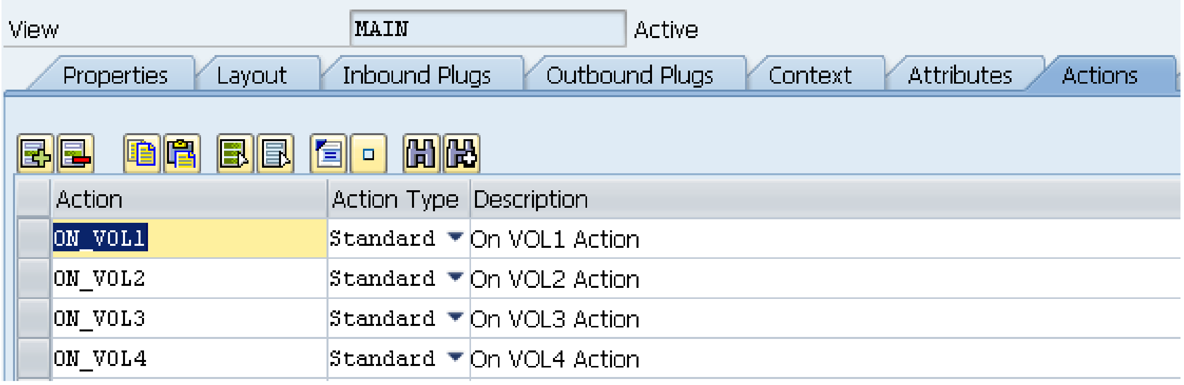
Do the following for the all 4 elements:


Create Action

Component	ZWDC_IMUNTORE_TREE
View	MAIN
Action	on_vol1
Description	On VOL1 Action

□ Transfer UI Event Parameters

Your actions tab must look like this:

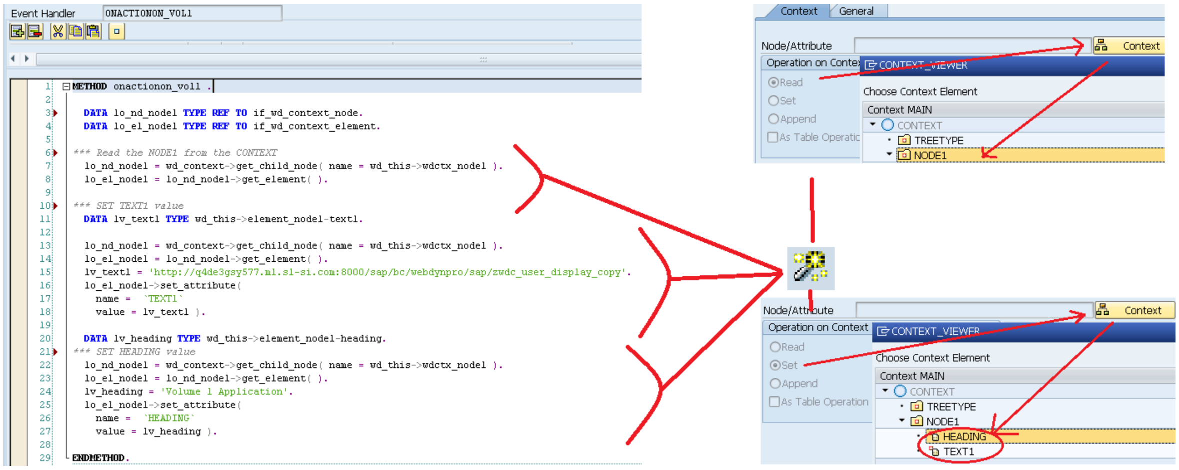


Action	Action Type	Description
ON_VOL1	Standard	On VOL1 Action
ON_VOL2	Standard	On VOL2 Action
ON_VOL3	Standard	On VOL3 Action
ON_VOL4	Standard	On VOL4 Action

7) Time to Code!

a. ON_VOL1 action:

Copy and past the code below. Remember that you can use the Wizard to do it:



```
METHOD onactionon_vol1 .

  DATA lo_nd_node1 TYPE REF TO if_wd_context_node.
  DATA lo_el_node1 TYPE REF TO if_wd_context_element.

*** Read the NODE1 from the CONTEXT
  lo_nd_node1 = wd_context->get_child_node( name = wd_this->wdctx_node1 ).
  lo_el_node1 = lo_nd_node1->get_element( ).

*** SET TEXT1 value
  DATA lv_text1 TYPE wd_this->element_node1-text1.

  lo_nd_node1 = wd_context->get_child_node( name = wd_this->wdctx_node1 ).
  lo_el_node1 = lo_nd_node1->get_element( ).
  lv_text1 = 'http://q4de3gsy577.ml.sl-si.com:8000/sap/bc/webdynpro/sap/zwdc_user_display_copy'.
  lo_el_node1->set_attribute(
    name =  `TEXT1`
    value = lv_text1 ).

  DATA lv_heading TYPE wd_this->element_node1-heading.
*** SET HEADING value
  lo_nd_node1 = wd_context->get_child_node( name = wd_this->wdctx_node1 ).
  lo_el_node1 = lo_nd_node1->get_element( ).
  lv_heading = 'Volume 1 Application'.
  lo_el_node1->set_attribute(
    name =  `HEADING`
    value = lv_heading ).

ENDMETHOD.
```

```
METHOD onactionon_vol1 .

  DATA lo_nd_node1 TYPE REF TO if_wd_context_node.
  DATA lo_el_node1 TYPE REF TO if_wd_context_element.

*** Read the NODE1 from the CONTEXT
  lo_nd_node1 = wd_context->get_child_node( name = wd_this-
>wdctx_node1 ).
  lo_el_node1 = lo_nd_node1->get_element( ).

*** SET TEXT1 value
  DATA lv_text1 TYPE wd_this->element_node1-text1.

  lo_nd_node1 = wd_context->get_child_node( name = wd_this-
>wdctx_node1 ).
```

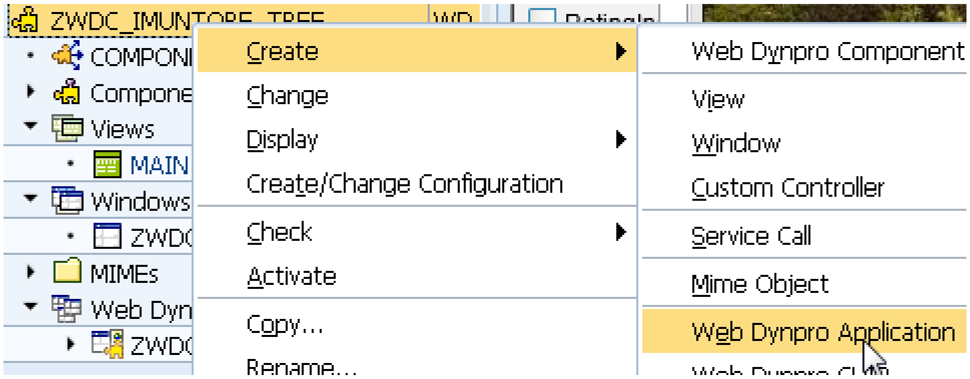
```
  lo_el_node1 = lo_nd_node1->get_element( ).
  lv_text1 = 'http://q4de3gsy577.m1.sl-
si.com:8000/sap/bc/webdynpro/sap/zwdc_user_display_copy'.
  lo_el_node1->set_attribute(
    name =  `TEXT1`
    value = lv_text1 ).

  DATA lv_heading TYPE wd_this->element_node1-heading.
*** SET HEADING value
  lo_nd_node1 = wd_context->get_child_node( name = wd_this-
>wdctx_node1 ).
  lo_el_node1 = lo_nd_node1->get_element( ).
  lv_heading = 'Volume 1 Application'.
  lo_el_node1->set_attribute(
    name =  `HEADING`
    value = lv_heading ).

ENDMETHOD.
```

b. ON_VOL2 action:
Just do the same but with lv_text1 = http://q4de3gsy577.m1.sl-si.com:8000/sap/bc/webdynpro/sap/zwdc_user_display_imuntor

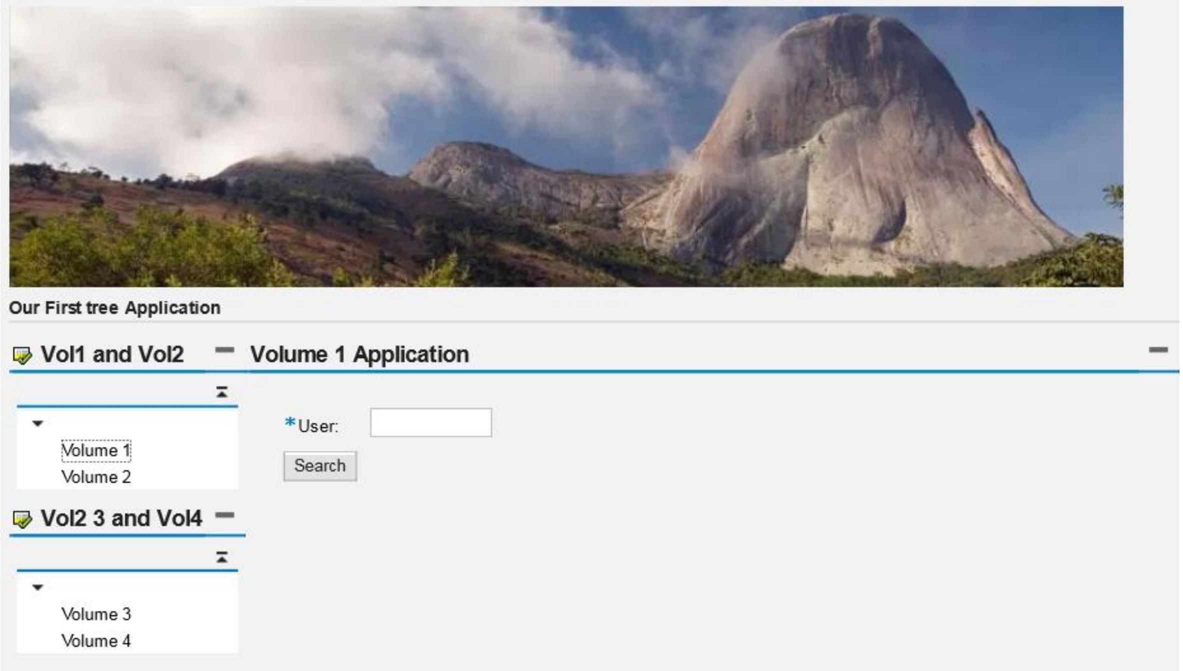
c. ON_VOL3 action:
Just do the same but with lv_text1 = http://q4de3gsy577.m1.sl-si.com:8000/sap/bc/webdynpro/sap/zwdc_imuntor_select_optio

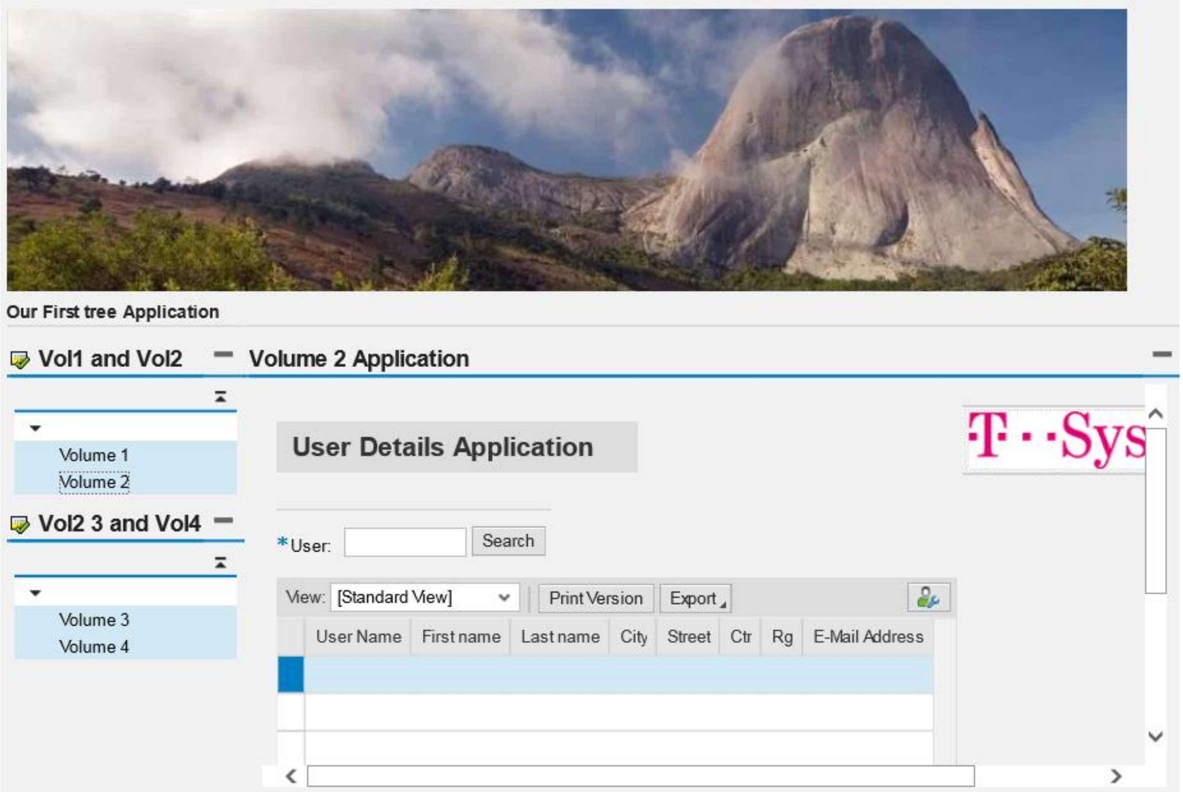
d. ON_VOL4 action:
Just do the same but with lv_text1 = http://q4de3gsy577.m1.sl-si.com:8000/sap/bc/webdynpro/sap/zwdc_imuntore_excel

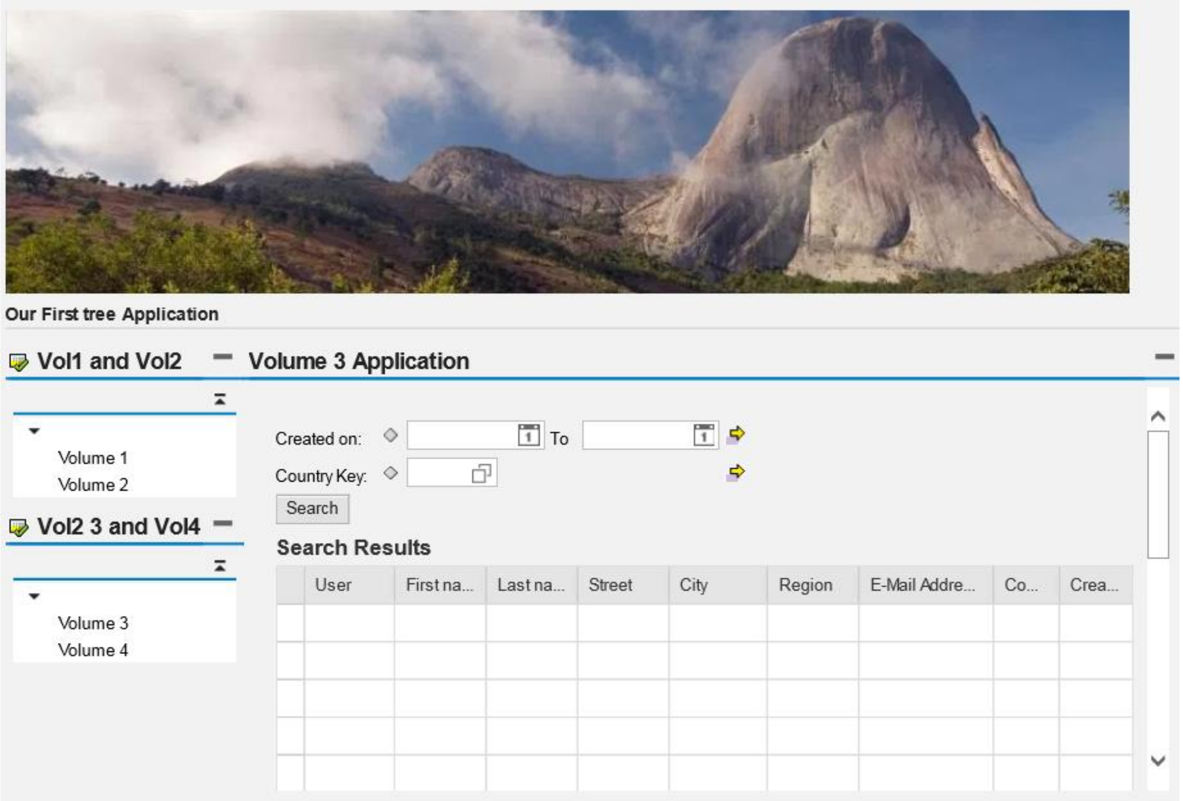
8) Create the WDA and execute the application:



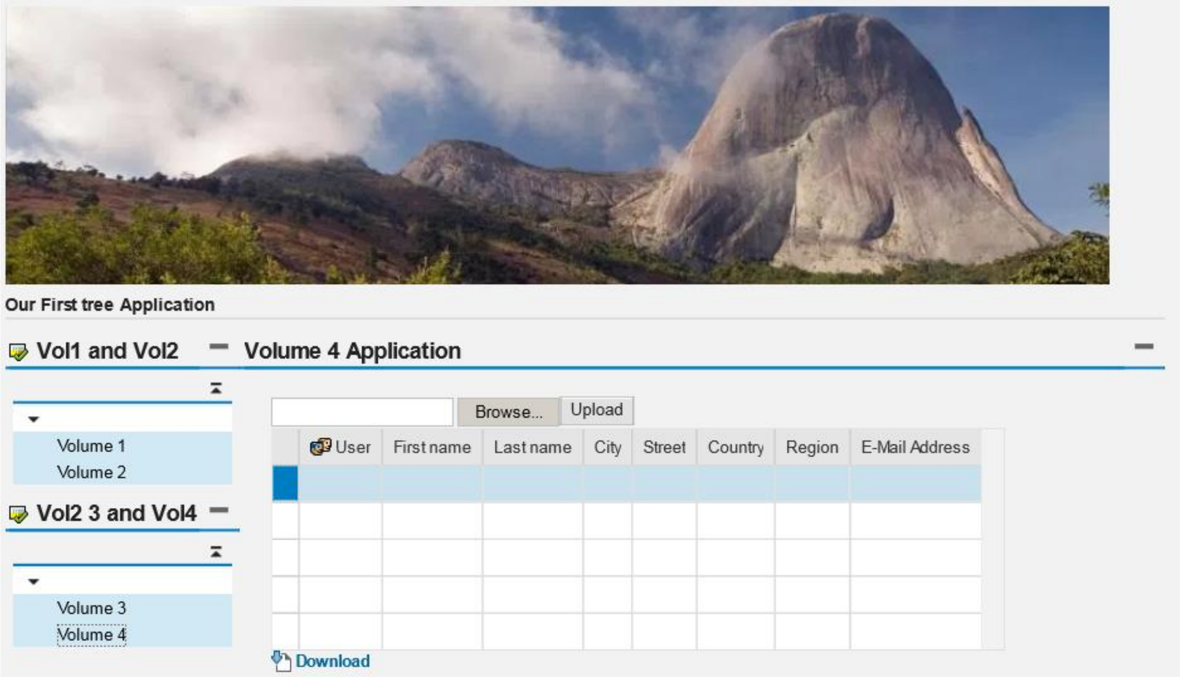
9) Final Result:







	User	First na...	Last na...	Street	City	Region	E-Mail Addre...	Co...	Crea...



	User	First name	Last name	City	Street	Country	Region	E-Mail Address