

Ejercicio 2 - Build a Dinamic Link – Orlando Sánchez Acuña

Part 1 – Add Local Variables to the Starter Form

ADV_DynamicAddress... DESIGN MODE EXPRESSION MODE TEST SAVE CHANGES appian

INTERFACE DEFINITION

```
1. a!localVariables(  
2.   local!locationsList: rule!ADV_getAddresses(addressId: NULL),  
3.   local!addingAddress: false(),  
4.   local!newAddress: type!{urn:com:appian:types:ADV}ADV_Address'(),  
5.  
6. {  
7.   a!sectionLayout(  
8.     label: "Address",  
9.     contents: {  
10.      a!radioButtonField(  
11.        label: "Radio Buttons",  
12.        labelPosition: "COLLAPSED",  
13.        choiceLabels:  
14.          a!forEach(  
15.
```

Address

- 358 Mayfair Dr. Unit #23 Lynn, MA 1902
- 358 Mayfair Dr. Unit #23 Lynn, MA 1902
- 47 New Saddle Ave. Unit # Goose Creek, SC 29445
- 47 New Saddle Ave. Unit # Goose Creek, SC 29445
- 7208 Oakland Drive Unit #50 Ravenna, OH 44266
- 7208 Oakland Drive Unit #50 Ravenna, OH 44266

RULE INPUTS

Name	Value
address	[addressId=1, shipp...

LOCAL VARIABLES

Name	Value
locationsList (6...	[addressId=3, shi...
addingAddress	false
newAddress	[addressId=, shipp...

allLocalVariables localVar1, localVarN, expression

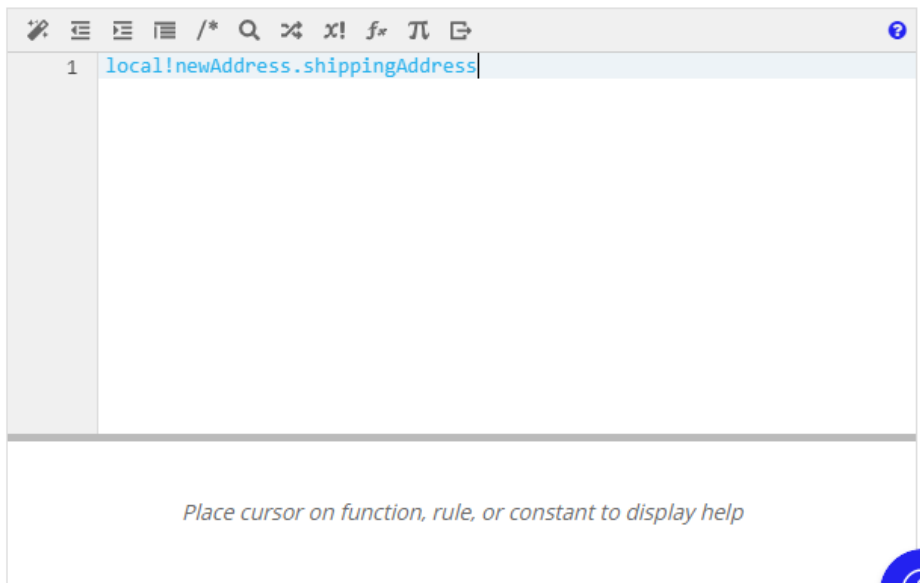
Lets you define one or more local variables for use within an expression. When used within an interface, the value of each variable can be refreshed under a variety of conditions, configured using `airefreshVariable()`. When used outside of an interface, all refresh properties configured using `airefreshVariable()` are ignored.

localVar1 (Any Type): The local variable to use when evaluating the given expression. Use the 'local' domain to define and reference individual variables. By default, a local variable will automatically update when any variables it references are changed. To change the way variables are updated, use the `airefreshVariable()` function. Variables can be refreshed under the following conditions: after each reevaluation, periodically on an interval, or when other variables change.

localVarN (Any Type): An unlimited number of local variables.

Open the Learning Center to select a tutorial or video

Text to display in the text field.



```
1 local!newAddress.shippingAddress
```

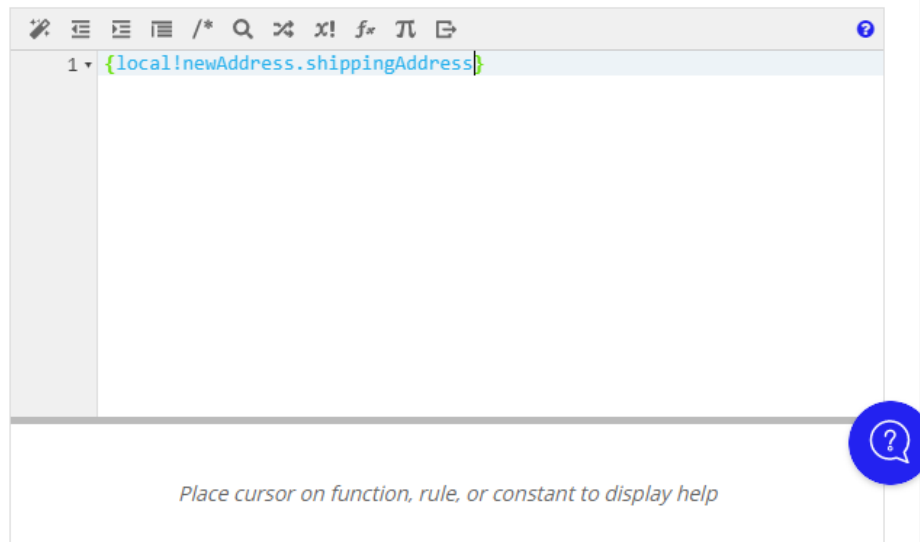
Place cursor on function, rule, or constant to display help

CANCEL

OK

Save Input To (List of Save)

One or more variables that are updated with the text when the user changes it. Use `alsave()` to save a modified or alternative value to a variable.

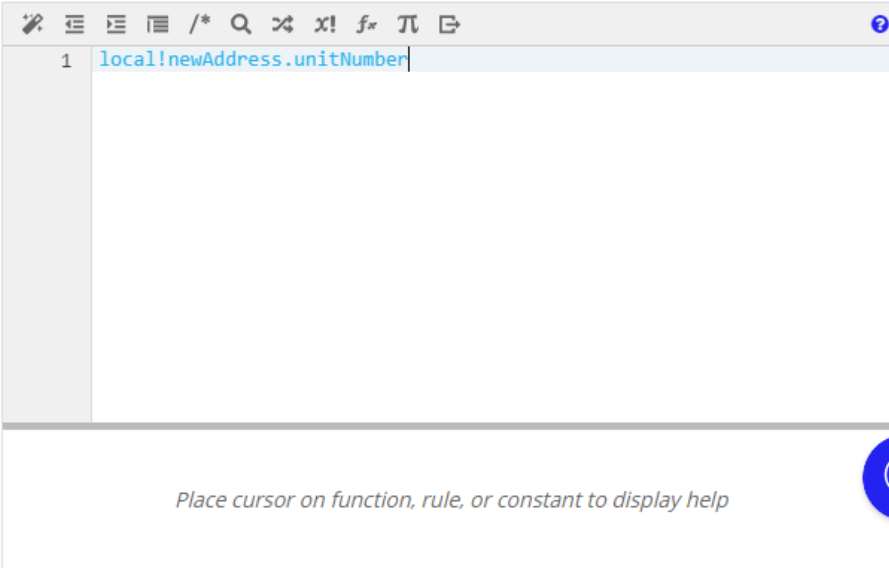


```
1 {local!newAddress.shippingAddress}
```

Place cursor on function, rule, or constant to display help

Display Value (Text)

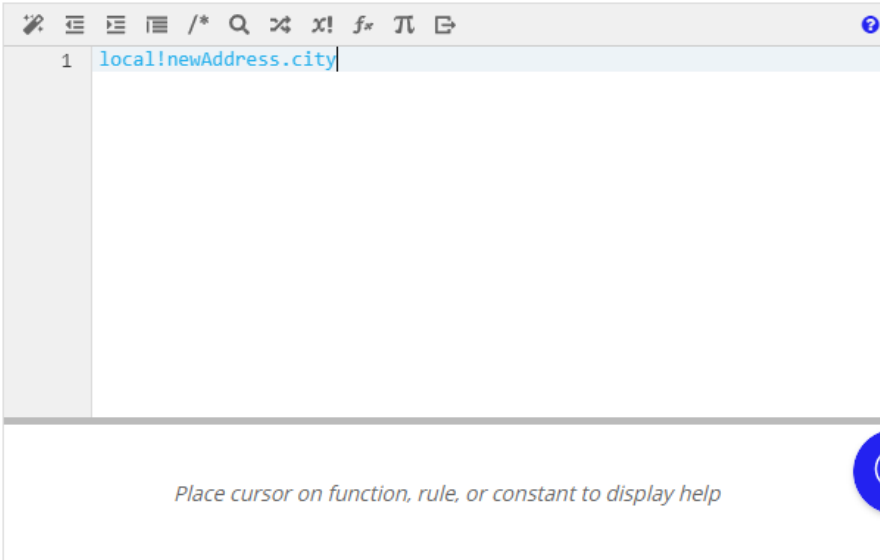
Text to display in the text field.



The screenshot shows a code editor interface. At the top, there is a toolbar with icons for undo, redo, search, and other editing functions. Below the toolbar, a text field contains the text `local!newAddress.unitNumber`. The text is highlighted in blue. Below the text field, there is a message that says "Place cursor on function, rule, or constant to display help" and a blue circular help icon with a question mark.

Display Value (Text)

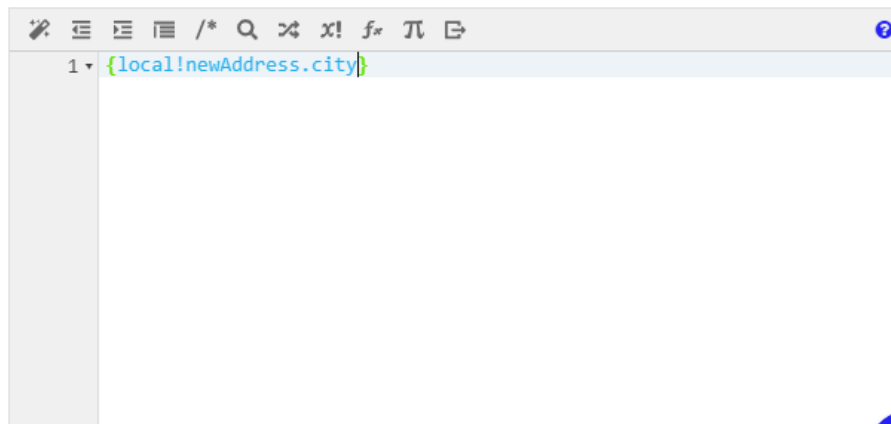
Text to display in the text field.



The screenshot shows a code editor interface. At the top, there is a toolbar with icons for undo, redo, search, and other editing functions. Below the toolbar, a text field contains the text `local!newAddress.city`. The text is highlighted in blue. Below the text field, there is a message that says "Place cursor on function, rule, or constant to display help" and a blue circular help icon with a question mark.

Save Input To (List of Save)

One or more variables that are updated with the text when the user changes it. Use `alsave()` to save a modified or alternative value to a variable.

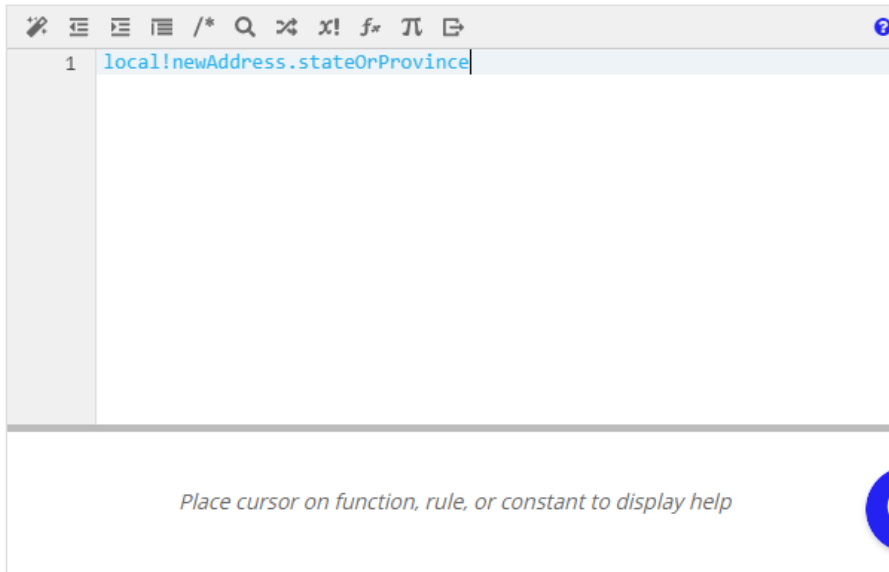


The screenshot shows a code editor window with a toolbar at the top containing icons for undo, redo, search, and other editing functions. The main area of the editor displays a single line of code: `{local!newAddress.city}`. The text is highlighted in blue, and a cursor is positioned at the end of the line. A small blue circle with a white question mark is visible in the bottom right corner of the editor window.

Place cursor on function, rule, or constant to display help

Display Value (Text)

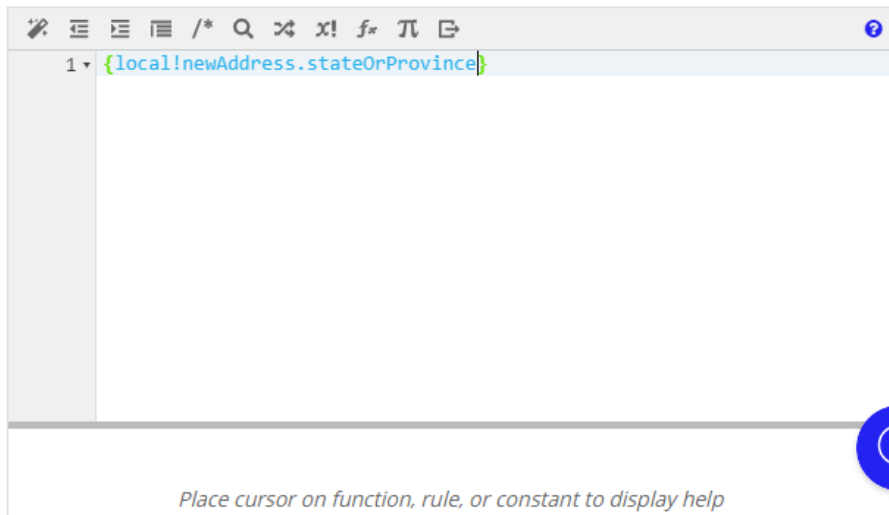
Text to display in the text field.



A screenshot of a code editor interface. The top toolbar contains icons for undo, redo, search, and other editing functions. The main editing area shows a single line of code: `1 local!newAddress.stateOrProvince`. The text is highlighted in blue. Below the editor, a message reads: *Place cursor on function, rule, or constant to display help*. A blue circular help icon with a question mark is located on the right side of the editor.

Save Input To (List of Save)

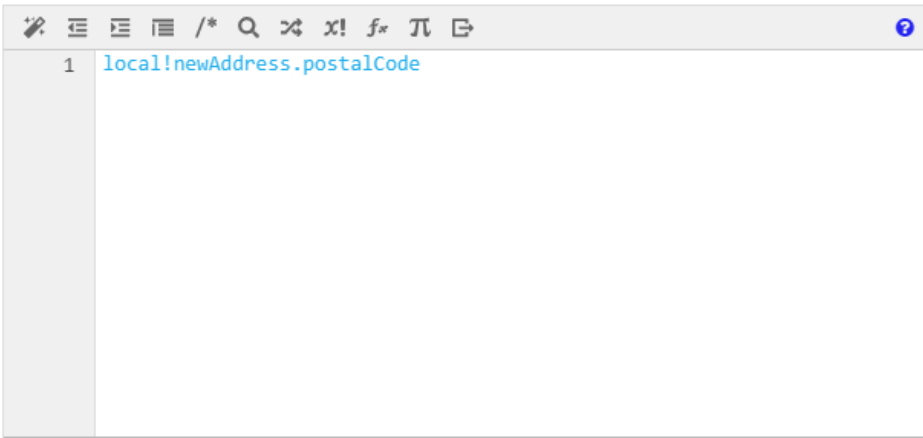
One or more variables that are updated with the text when the user changes it. Use `alsave()` to save a modified or alternative value to a variable.



A screenshot of a code editor interface, similar to the one above. The main editing area shows a single line of code: `1 {local!newAddress.stateOrProvince}`. The text is highlighted in blue. Below the editor, a message reads: *Place cursor on function, rule, or constant to display help*. A blue circular help icon with a question mark is located on the right side of the editor.

Display Value (Text)

Text to display in the text field.



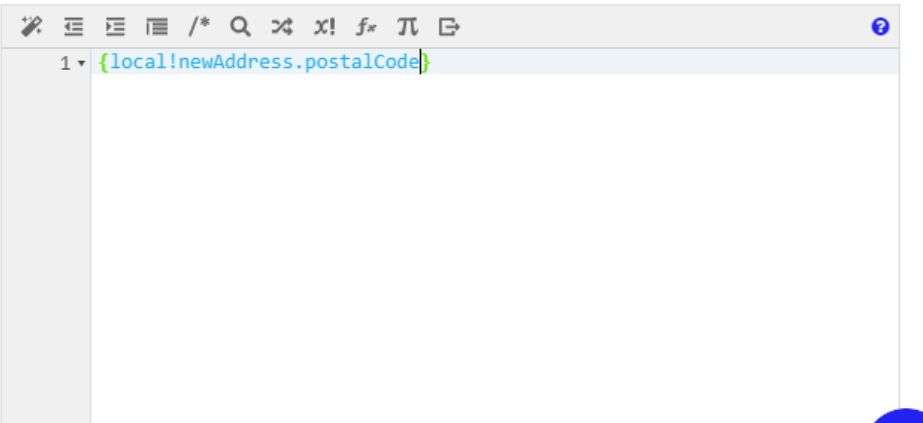
```
1 local!newAddress.postalCode
```

Place cursor on function, rule, or constant to display help



Save Input To (List of Save)

One or more variables that are updated with the text when the user changes it. Use `alsave()` to save a modified or alternative value to a variable.



```
1 {local!newAddress.postalCode}
```

Place cursor on function, rule, or constant to display help



The image displays two screenshots of the Appian ADV_DynamicAddress... interface, illustrating the configuration of an 'Add New Address' form.

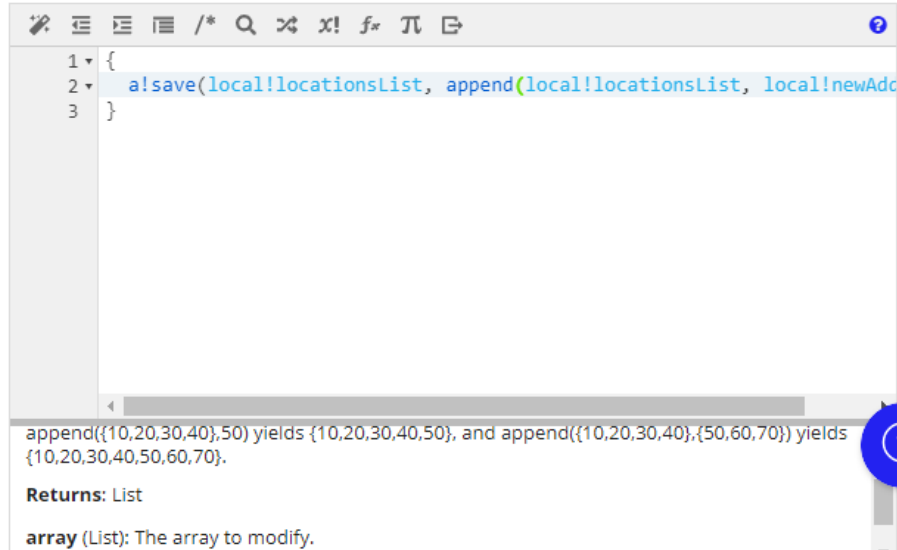
Top Screenshot: The form is titled 'Address' and contains a list of radio buttons for address selection. The selected option is '47 New Saddle Ave. Unit # Goose Creek, SC 29445'. Below the list is the 'Add New Address' section, which includes a 'Shipping Address *' field with the value '5555', a 'Unit Number' field, a 'City *' field with the value 'Chicago', a 'State/Province' field with the value 'US', and a 'Post Code' field with the value '222'. The configuration pane on the right shows the 'Text' component selected under 'COMPONENT CONFIGURATION'.

Bottom Screenshot: The form is identical to the top screenshot, but a blue 'Button' component with the text 'ADD' is now visible at the bottom right of the 'Add New Address' section. The configuration pane on the right shows the 'Button' component selected under 'COMPONENT CONFIGURATION'.

Part 4 – Configure the Addition and Selection of the New Address

Save Value To (List of Save)

One or more variables that are updated with the button value when the user presses it. Use `alsave()` to save a modified or alternative value to a variable.



```
1 {  
2   alsave(local!locationsList, append(local!locationsList, local!newAdd  
3 }
```

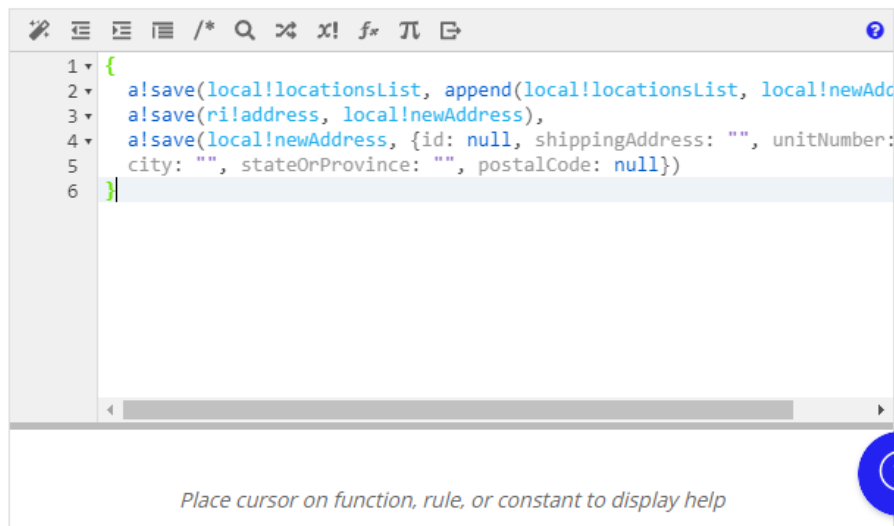
`append({10,20,30,40},50)` yields `{10,20,30,40,50}`, and `append({10,20,30,40},{50,60,70})` yields `{10,20,30,40,50,60,70}`.

Returns: List

array (List): The array to modify.

Save Value To (List of Save)

One or more variables that are updated with the button value when the user presses it. Use `alsave()` to save a modified or alternative value to a variable.



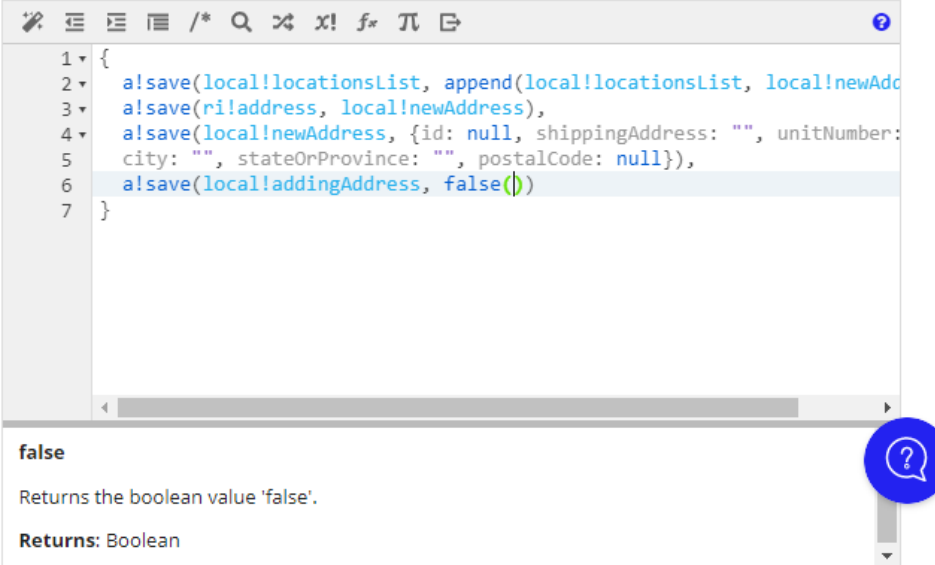
```
1 {  
2   alsave(local!locationsList, append(local!locationsList, local!newAdd  
3   alsave(ri!address, local!newAddress),  
4   alsave(local!newAddress, {id: null, shippingAddress: "", unitNumber:  
5   city: "", stateOrProvince: "", postalCode: null})  
6 }
```

Place cursor on function, rule, or constant to display help

Part 5 – Set the addingAddress Flag to False

Save Value To (List of Save)

One or more variables that are updated with the button value when the user presses it. Use `alsave()` to save a modified or alternative value to a variable.



```
1 {
2   alsave(local!locationsList, append(local!locationsList, local!newAdd
3   alsave(ri!address, local!newAddress),
4   alsave(local!newAddress, {id: null, shippingAddress: "", unitNumber:
5   city: "", stateOrProvince: "", postalCode: null}),
6   alsave(local!addingAddress, false())
7 }
```

false

Returns the boolean value 'false'.

Returns: Boolean

Part 6 – Hide the Add New Address Form

Visibility (Boolean)

Determines whether the layout is displayed on the interface. When set to false, the layout is hidden and is not evaluated. Default: true.



```
1 local!addingAddress
```

Place cursor on function, rule, or constant to display help

Part 7 – Show the Add New Address Form

ADV_DynamicAddressList DESIGN MODE EXPRESSION MODE TEST SAVE CHANGES appian

COMPONENTS PATTERNS Search Components...

- DATE
- DATE & TIME
- ENCRYPTED TEXT
- FILE UPLOAD
- SIGNATURE
- BARCODE
- SELECTION
 - DROPDOWN
 - MULTIPLE DROPDOWN
 - RADIO BUTTONS
 - CHECKBOXES
 - CARD CHOICES
- DISPLAY
 - IMAGE
 - RICH TEXT
 - TAG
 - STAMP

Address

- 358 Mayfair Dr. Unit #23 Lynn, MA 1902
- 358 Mayfair Dr. Unit #23 Lynn, MA 1902
- 47 New Saddle Ave. Unit # Goose Creek, SC 29445
- 47 New Saddle Ave. Unit # Goose Creek, SC 29445
- 7208 Oakland Drive Unit #50 Ravenna, OH 44266
- 7208 Oakland Drive Unit #50 Ravenna, OH 44266

Rich Text # asdasd.

+ Add New Address

Add New Address

Shipping Address *		Unit Number
<input type="text"/>	<input type="text"/>	<input type="text"/>
City *	State/Province	Post Code
<input type="text"/>	<input type="text"/>	<input type="text"/>

ADD

RULE INPUTS

Name	Value
address	[addressId+, shipp...

LOCAL VARIABLES

COMPONENT CONFIGURATION

Rich Text

Label

Label Position
Hidden

Instructions

Help Tooltip
Shows next to label

Display Value