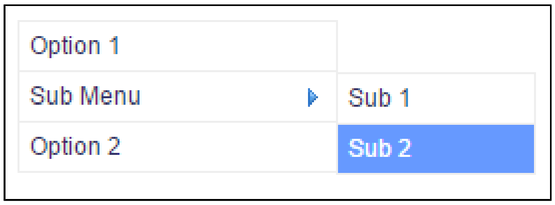
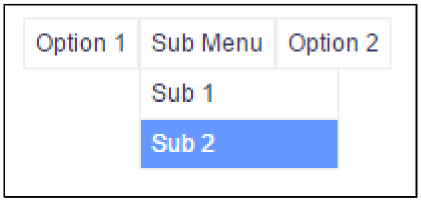
Menu

Overview

The menu widget is used to create navigation or context menus. Menu options are populated using the 'choices' and 'choice values' properties, and the user's selection is received in the application using the 'menu response' property.





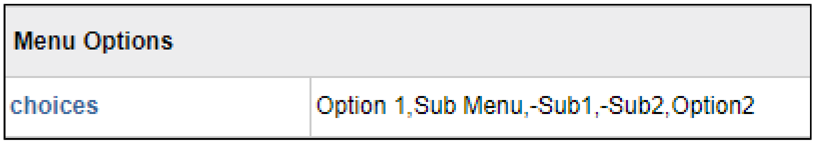
The menu widget is a control that is placed within an application screen and the user's selection is processed by the application itself. The menu widget is not, by itself, a device for calling other programs directly. For a menu system that is designed to call programs when the user makes a selection, see Atrium.

Field Binding Dialog (Rich UI Only)

Please visit the Field Binding page.

Menu Options

The menu options are specified using the 'choices' and 'choice values' properties. 'choices' specifies the option display text, and 'choice values' specifies the option return value. To specify multiple options, both properties accept a comma-separated list. To create sub-menus, items can be prefixed with a dash ('-') or multiple dashes to indent the items. For example, to create a menu as shown in the screenshots above:



Menu Options	
choices	Option 1,Sub Menu,-Sub1,-Sub2,Option2

- In Rich Display Files, the 'choices' and 'choice values' properties can be bound for dynamic options.
- If 'choice values' are not specified for each 'choice', then the 'choice' text will be returned to the program as-is.
- If more options are specified than will fit in the widget, scroll bars will appear as necessary. The scroll bars can be disabled by setting the 'overflow x' and 'overflow y' properties to 'hidden'.

Mouse-over Animation

By default, the mouse-over effect is animated, set the 'animate' property to 'false' to disable this behavior.

Menu Orientation

By default, the menu will display vertically. To display the menu horizontally, set the 'orientation' property to 'horizontal'.

Menu Response (Rich UI Only)

When the 'menu response' property is bound to a field, clicking on a menu option will submit the screen to the application for processing. The field bound to 'menu response' will be populated with the 'choice values' (if specified) or 'choice' text associated with the menu item.

onoptionclick Event

To process the user's menu selection using JavaScript, the "onoptionclick" event property can be used. This event fires when the user clicks on a menu option. Variables "text" and "value" are made available to the event handler, these contain the clicked menu option text and value. For example, the following code could be used with "onoptionclick" to process a menu selection in Genie:

```
pui.set("I_20_7", value); // Write option value into field at 20,7
pressKey("enter");
```

Menu option icons

To configure menu option icons, the following properties have to be set:

- "menu option indent" - set this property to a value that would allow space for the icon (for example: 20px). Select "Other..." in the dropdown if the value you're looking for is not available.
- "background repeat" - select no-repeat. Otherwise, the icon will repeat horizontally.
- "option image" - to specify a different icon for each menu option, right-click the property, select open editor so that you have more space, and then enter a comma separated list of image paths each one corresponding to appropriate menu option in your choices list.

Use as Grid Context Menu (Rich UI Only)

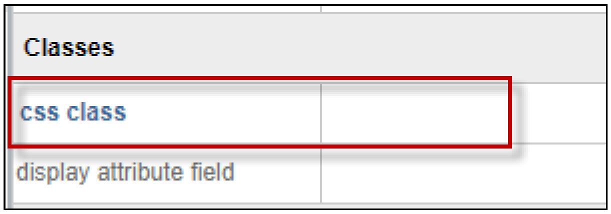
To use a menu widget as a grid context menu:

- Add a menu widget to the display and hide it by setting the "visibility" property to "hidden".
- Set the grid's "context menu id" property to the id of the menu widget.

The menu widget will be automatically shown when the user right-clicks on a grid record, and hidden when the user clicks on the screen again. The grid's onrowclick event property can be used in conjunction with the applyProperty() API to set/reset the menu options if the options need to be specific to a grid row.

CSS

Menu styling rules happen in a CSS class named, "pui-menu". You can write CSS in a custom stylesheet and apply it to a menu using the "css class" menu property. See CSS Class for details.



If you specify a class, then your className is added to the class list along with "pui-menu". Horizontally oriented menus have an additional class, "pui-horizontal". You can override the default rules by adding a custom stylesheet to /profoundui/userdata/custom/css/ (where profoundui is your Profound UI installation name). Specify rules for whichever of these selectors you'd like to change:

```
div.pui-menu {}
div.pui-menu.pui-horizontal {}
div.pui-menu table {}
div.pui-menu table td {}
```

Hover Class Name

If the hover background color property is not set, then hovering over a menu option causes its class name to become "menu-hover". That allows you to define your own custom CSS for an option when it is hovered over.