

ColumnOption Class

window.almworks.structure.api.ColumnOption

ColumnOption class represents a single column configuration parameter.

It needs to be subclassed for particular column type implementation and passed as return value in [ColumnConfigurator.getOptions\(\)](#) method.

Options are displayed in column configuration dialog one after another with labels on the left and inputs on the right.

Example

```
var api = window.almworks.structure.api;

var MyOption1 = api.subClass('MyOption', api.ColumnOption, {
  title: 'Some option',
  init: function() {
    this.input$ = null;
  },
  createInput: function(div$) {
    this.input$ = div$.append('<input type="text" class="text">').find('input');
    var params = this.spec.params;
    this.input$.on('change', function() {
      if (params.someOptionAvaible) {
        params.someOption = $(this).val();
        div$.trigger('notify');
      }
    });
  },
  notify: function() {
    var available = this.spec.params.someOptionAvaible;
    this.input$.val(available ? (this.spec.params.someOption || '42') : '');
    return available;
  }
});
```

Properties

title

If set, title is displayed as a label to the left of the input controls. Option title representation may be overridden in [#createLabel\(div\\$\)](#) method.

Required Methods

You need to override the following methods.

createInput(div\$)

Should be overridden to provide custom HTML for the option input. `div$` parameter provides parent option element to append your view to. Created input should trigger 'notify' event on `div$` to notify Structure of any column parameters change.

Please honor the AUI Forms HTML layout when creating your input controls!

Example

```

createInput: function(div$) {
  var self = this;
  this.input$ = $('<input type="text" class="text">').appendTo(div$).on('change', function() {
    if (self.spec.params.myOption !== $(this).val()) {
      self.spec.params.myOption = $(this).val();
      div$.trigger('notify');
    }
  });
}

```

Other Methods

init(options)

Optional initializer.

createLabel(div\$)

May be overridden to provide custom HTML view for the input label. `div$` parameter provides parent option element to append your view to. By default creates a right-aligned label with text of the `#title` property.

Please honor the AUI Forms HTML layout if you override this method!

notify()

This method is called when the column configuration has changed. The implementation may want to update its controls to reflect those changes. The method should return a `boolean` indicating whether this option is available. Unavailable options will not be shown on the configuration panel. The default implementation does nothing and always returns `true`.

Example

```

notify: function() {
  this.input$.val(this.spec.params.myOption);
  return true;
}

```

isInputValid()

Returns `true` if the current column specification is valid from the point of view of this option. The column configuration won't be saved unless all of the options approve the specification. The default implementation does nothing and returns `true`.

Example

```

isInputValid: function() {
  // Check that the "field" specification parameter is present.
  return !!this.spec.params.field;
}

```