

•  
•  
•



button\_group

```
,,, htape,,  
.BasicButton,,, stopPropagation: true  
,?fineui, EVENT_DELETE . Fixcontext,,
```

```

!(function () {

    var Item = BI.inherit(BI.BasicButton, {

        props: {
            baseCls: "my-cls",
            text: "",
            height: 24
        },

        render: function () {
            var self = this, o = this.options;
            return {
                type: "bi.htape",
                items: [
                    {
                        el: {
                            type: "bi.label",
                            text: o.text,
                            textAlign: "left"
                        },
                        el: {
                            type: "bi.checkbox",
                            ref: function (_ref) {
                                self.checkbox = _ref;
                            },
                            height: 24
                        },
                        width: 24
                    },
                    {
                        el: {
                            type: "bi.icon_button",
                            cls: "delete-node-font",
                            stopPropagation: true,
                            handler: function () {
                                self.fireEvent("EVENT_DELETE", o.value);
                            }
                        },
                        width: 24
                    }
                ]
            };
        }

        // ,button_groupitemssetSelected,isSelected,getValue,BasicButton,,,setSelected
        setSelected: function (b) {
            Item.superclass.setSelected.apply(this, arguments);
            this.checkbox.setSelected(b);
        }
    });

    BI.shortcut("dec.demo.item", Item);
}());

```

button\_group.

ListitemsEVENT\_DELETE,

populate.

button\_group

```

!(function () {

    var List = BI.inherit(BI.Widget, {

        props: {

```

```

baseCls: "dec-frame-body",
items: [
    {
        text: "1",
        value: 1
    },
    {
        text: "2",
        value: 2
    }
],
};

render: function () {
    var self = this, o = this.options;

    return {
        type: "bi.vertical",
        items: [
            {
                el: {
                    type: "bi.button",
                    text: "",
                    handler: function () {
                        self._addOne();
                    }
                }
            },
            {
                el: {
                    type: "bi.button_group",
                    ref: function (_ref) {
                        self.list = _ref;
                    },
                    layouts: [
                        {
                            type: "bi.vertical"
                        }
                    ],
                    chooseType: BI.Selection.Single,
                    scrollBy: true,
                    items: this._formatItems(o.items),
                    listeners: [
                        {
                            eventName: "EVENT_CHANGE",
                            action: function () {
                                self._handleSelect();
                            }
                        }
                    ]
                }
            }
        ];
    };
}

_addOne: function () {
    this.options.items.push({
        text: BI.randomUUID(),
        value: BI.randomUUID()
    });
    this.populate(this.options.items);
},

_deleteOne: function (value) {
    this.options.items = BI.filter(this.options.items, function (index, item) {
        return item.value !== value;
    });
    this.populate(this.options.items);
},

_handleSelect: function () {
    var selectedValues = this.getValue();
    BI.each(this.options.items, function (index, item) {
        item.selected = BI.contains(selectedValues, item.value);
    });
}

```

```
        });
    },

    _formatItems: function (items) {
        var self = this;
        return BI.map(items, function (index, item) {
            return BI.extend({
                type: "dec.demo.item",
                listeners: [
                    {
                        eventName: "EVENT_DELETE",
                        action: function (value) {
                            self._deleteOne(value);
                        }
                    }
                ]
            }, item);
        });
    },

    getValue: function () {
        return this.list.getValue();
    },

    populate: function (items) {
        this.list.populate(this._formatItems(items));
    }
});

BI.shortcut("dec.demo", List);
}());

```

