Keypad

Component for entering data without a keyboard. Excellent to create the element for pin enter.

About

In Metro 4, creating a keypad is very simple. To create `keypad` add attribute `data-role="keypad"` to `<input>` element.

| Enter pin | Enter pin |

```
<input type="text" data-role="keypad" placeholder="Enter pin">
<input type="password" data-role="keypad" placeholder="Enter pin">
```

Options

The streamer contains a number of options for defining behavior:

Option	Data-*	Default	Desc
keySize	`data-key-size`	32	Can be a number. Key button dimension in pixels
keys	`data-keys`	1, 2, 3, 4, 5, 6, 7, 8, 9, 0	Can be a string. The character set. This a string with a comma delimiter.
target	`data-target`	null	Selector (class or id) for additional target for value.
keyLength	`data-key-length`	0	Can be a number. This is an internal constraint. If value more than 0, a user can't enter a value with length more then length value.
shuffle	`data-shuffle`	false	Can be true or false. If a value for this option is true, keys can be shuffled after each key entering.
shuffleCount	`data-shuffle-count`	3	Can be number. The number of shuffling of the array of characters
position	`data-position`	bottom-left	Keys position. Can be a: left, top-left, top, top-right, right, bottom-right, bottom, bottom-left.
dynamicPosition	`data-dynamic-position`	false	Can be true or false. If true - keys wrapper change position after key click.
serviceButtons	`data-service-buttons`	true	Can be true or false. If a value for this option is a false, service buttons (backspace, clear) will not be drawing.
showValue	`data-show-value`	true	Can be true or false. If a value for this option is a false, the result value will not be drawing on the input element.
open	`data-open`	false	Can be true or false. If a value for this option is a true, a keys showing always.
sizeAsKeys	`data-size-as-keys`	false	Can be true or false. If a value for this option is a true, a size of inputs set to size equal to keys wrapper size.
clsKeypad	`data-cls-keypad`		Additional class for keypad
clsInput	`data-cls-input`		Additional class for input field
clsKeys	`data-cls-keys`		Additional class for keys wrapper
clsKey	`data-cls-key`		Additional class for each key
clsServiceKey	`data-cls-service-key`		Additional class for each service key
clsBackspace	`data-cls-backspace`		Additional class for backspace key
clsClear	`data-cls-clear`		Additional class for clear key

Below, some options will be discussed in more detail.

Character set

By default keypad contains numbers from 0 to 9. You can change default character set with attribute `data-keys`.

| Default set | Custom set |
| Enter pin | Enter pin |

```
<input type="text" data-role="keypad"
    data-keys="q,w,e,r,t,y,u,i,o,p,a,s,d,f,g,h,j,k,l,z,x,c,v,b,n,m">
```

Keys position

By default keys showing on bottom-left from input. You [can]... with attribute `data-position`.

1	2	3
4	5	6
7	8	9
0	←	×

○ bottom-left ○ bottom ○ bottom-right ○ to...
○ top-right ○ right ● left

Also you can use attribute `data-dynamic-position` to change keys position after user key clicked.

```
<input type="text" data-role="keypad" data-dynamic-position="true">
```

Additional target for value

You can set **additional** target for `value` . To set additional target for value use attribute `data-target` with target selector. **Selector** - a string containing a selector expression to match elements against.

```
Enter pin                                    Mirror pin
```

```
<input type="text" data-role="keypad" data-target="#keypad_target">
<input type="text" id="keypad_target" readonly>
```

Constraints

You can set attribute `data-key-length` to limit the number of characters to be entered.

```
Enter the six-digit pin
```

```
<input type="text" data-role="keypad"
       placeholder="Enter the six-digit pin" data-key-length="6">
```

Shuffle

You can use attribute `data-shuffle` to change keys data position after each key entering.

```
<input type="text" data-role="keypad" data-shuffle="true">
```

Also you can combine `shuffle` action with `dynamic position` .

```
<input type="text" data-role="keypad"
       data-shuffle="true" data-dynamic-position="true">
```

Customize

Component `keypad` contains special options for customizing. To customize keypad use options with prefix `cls` or relevant attributes `data-cls-*` .

```
<input type="text"
       data-role="keypad"
       data-position="top"
       data-cls-keypad="info"
       data-cls-input="bg-green fg-white"
       data-cls-keys="bg-cyan fg-white"
       data-cls-backspace="bg-darkOrange fg-white"
       data-cls-clear="bg-darkRed fg-white">
```

Events

When `keypad` works, it generated the numbers of events. You can use callback for this event to behavior with component.

Event	Data-*	Desc
onChange(val, el)	`data-on-change`	Fired when value changed
onClear(el)	`data-on-clear`	Fired when user click clear button
onBackspace(value, el)	`data-on-backspace`	Fired when user click backspace button
onShuffle(new_keys_order, keys, el)	`data-on-shuffle`	Fired when keys is shuffling
onKey(key, value, el)	`data-on-key`	Fired when user click key
onKeypadCreate(el)	`data-on-keypad-create`	Fired when keypad is created

```
Enter a pin                                  Pin is:
```

```
<div class="row">
    <div class="cell-md-6">
        <input placeholder="Enter a pin"
        type="text" data-role="keypad"
        data-on-change="$('#change_target').val(arguments[0])">
    </div>
    <div class="cell-md-6">
        <input type="text" data-role="input"
        id="change_target" data-prepend="Pin is: "
        data-clear-button="false" readonly>
    </div>
```

```
    </div>
```

Methods

Component `keypad` contains method to interact with keypad.

1	2	3
4	5	6
7	8	9
0	←	×

- **val(v)** - get or set value
- **open()** - open keys panel
- **close()** - close keys panel
- **setPosition(pos)** - set new position
- **shuffleKeys(iteration)** - shuffle

Shuffle keys

```
<div class="row">
    <div class="cell-md-6">
        <button class="button"
        onclick="$('#keypad_methods').data('keypad').shuffleKeys(10)">
            Shuffle keys
        </button>
    </div>
    <div class="cell-md-6">
        <input data-role="keypad" id="keypad_methods"
            data-open="true" data-position="top">
    </div>
</div>
```

Additional classes

You can use additional classes to change input size: `.input-large` and `.input-small` .

Input small

Input regular

Input large