



Search...

Getting started
Components

- M4Q
- About
- Population
- Constructor
- Ajax
- Animation
- Loops
- Visibility
- Effects
- Subtree functions
- Attributes
- Html, text and value
- Css and classes
- Position and size
- Manipulation
- DataSet
- Events
- Utils
- Base
- Containers
- Grid system
- Typography
- Tables
- Forms
- Buttons
- Images
- Figures
- Lists
- Form components
- Checkbox
- File input
- Input
- Input material
- Keypad
- Rating
- Radio
- Select
- Slider
- Double Slider**
- Spinner
- Switch
- Tag input
- Textarea

- Menus
- App bar
- Bottom navigation
- Bottom sheet
- Menu
- Ribbon Menu
- Side bar
- Side navigation

- Controls
- Accordion
- Badge
- Carousel
- Cards
- Cube
- Counter
- Charms
- Chat
- Donut
- Image compare
- Image magnifier
- Gravatar
- List
- ListView
- Master
- NavigationView
- Panels
- Progress & Activity
- Streamer
- Stepper
- Splitter
- Tabs
- Tabs material
- Table
- Tiles
- TreeView
- Wizard

- Information
- Dialogs
- Info box
- Hints
- Notify system
- Popovers
- Toasts
- Windows

- Date & time
- Calendar
- Calendar picker
- Date picker
- Time picker
- Countdown
- Formatting date


- Media

Double Slider

Let the user specify a numeric range value with a double slider component.

About slider

Component `doubleslider` let the user specify a numeric range value which must be no less than a given value, and no more than another given value. To create `doubleslider` add attribute `data-role="doubleslider"` to input element.



```
<input data-role="doubleslider">
```

Currently Metro support only horizontal type of slider.

Slider value

The slider return **comma separated** value.

Return actual value



```
<input data-role="doubleslider" data-min="-40" data-max="40"
  onchange="$('#slider-return-value').val(this.value)">

<input type="text" id="slider-return-value">
```

You can change attribute `data-value-min` , `data-value-max` at runtime and slider will be updated.

Accuracy

You can set `accuracy` for slider with attribute `data-accuracy` .



```
<input data-role="doubleslider" data-accuracy="5" data-hint-always="true">
```

Additional target


You can put slider value to additional targets. To set it, add attribute `data-target="..."` to element. Value for this attribute must be selector specific string.



```
<input data-role="doubleslider" data-target="#slider-target1, #slider-target2">
<div class="p-2 bg-cyan fg-white text-center" id="slider-target"></div>
<input type="text" id="slider-target2">
```

Hint

You can enable `hint` for slider. To enable `hint` add attribute `data-hint="true"` to element.



```
<input data-role="doubleslider" data-hint="true">
```


Also you can make the hint as permanent. To set `hint` in `permanent` mode add attribute `data-hint-always="true"` .



```
<input data-role="doubleslider" data-hint="true" data-hint-always="true">
```

Hint position

You can set `hint position` with attribute `data-hint-position-min` and `data-hint-position-max` . To set specific position use next values for this attribute: `top` , `bottom` , `left` and `right` .



```
<input data-role="doubleslider" data-value-min="20">
```

Table of contents

- Slider
- About slider
- Slider value
- Accuracy
- Additional target
- Hint
- Position
- Value
- Options
- Events
- Methods
- Observe
- Customize

```
data-value-max="80"  
data-hint-position-min="left"  
data-hint-position-max="right"  
data-hint-always="true"  
data-cls-hint="bg-red fg-white">
```

Hint value

You can use `template` for hint value with two variables `$1`.

0

Min: 20

Max: 80

100

```
<input data-role="doubleslider"  
  data-show-min-max="true"  
  data-hint-mask-min="Min: $1"  
  data-hint-mask-max="Max: $1"  
  data-hint-always="true"  
  data-value-min="20"  
  data-value-max="80"  
  data-cls-hint="bg-cyan fg-white">
```

Options

Option	Data-*	Default	Description
roundValue	<code>data-round-value</code>	<code>true</code>	Round range value
min	<code>data-min</code>	<code>0</code>	Slider min value
max	<code>data-max</code>	<code>100</code>	Slider max value
accuracy	<code>data-accuracy</code>	<code>0</code>	Slider accuracy value
showMinMax	<code>data-show-min-max</code>	<code>false</code>	Show min/max slider values
minMaxPosition	<code>data-min-max-position</code>	<code>top</code>	Where shown min/max values. Can top, bottom
valueMin	<code>data-value-min</code>	<code>null</code>	Slider minimum value. If not defined, used min
valueMax	<code>data-value-max</code>	<code>null</code>	Slider maximum value. If not defined, used max
hint	<code>data-hint</code>	<code>false</code>	Show/hide hint on marker
hintAlways	<code>data-hint-always</code>	<code>false</code>	Show hint always

Events

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

onStart(val, percent, slider)	<code>data-on-start</code>	Fired when start sliding
onStop(val, percent, slider)	<code>data-on-stop</code>	Fired when stop sliding
onMove(val, percent, slider)	<code>data-on-move</code>	Fired when user move slider marker
onChange(val, percent)	<code>data-on-change-value</code>	Fired when slider value was changed
onChangeValue(val)	<code>data-on-change-value</code>	Fired when slider value was changed
onFocus(val, percent, slider)	<code>data-on-focus</code>	Fired when slider marker get focus
onBlur(val, percent, slider)	<code>data-on-blur</code>	Fired when slider marker loses focus
onDoubleSliderCreate(slider)	<code>data-on-slider-create</code>	Fired when slider was created

Also you can use standard `onchage` event for `input` with `data-role="slider"`.

-100

100

```
<input data-role="slider"  
  data-show-min-max="true"  
  data-min="-100" data-max="100"  
  data-on-change-value="$('#event-receiver').val('Value: '+arguments[0])">  
  
<input type="text" id="event-receiver">
```

Methods

In additional, You can use slider methods to interact with the component.

- **val()** - get value
- **val(vMin, vMax)** - set value

```
<div class="row">  
  <div class="cell-md-6">  
    <input type="text" min="0" max="100"  
      oninput="$('#slider-methods').data('slider').val(this.value)">
```

```
</div>
<div class="cell-md-6">
  <input data-role="slider" data-accuracy="10" id="slider-methods">
</div>
</div>
```

Customize

If you need to customize the slider, you can use next options:

Option	Data-*	Desc
clsSlider	data-cls-slider	Additional class for slider
clsBackside	data-cls-backside	Additional class for slider backside
clsComplete	data-cls-complete	Additional class for slider complete
clsMarker	data-cls-marker	Additional class for slider marker
clsMarkerMin	data-cls-marker-min	Additional class for slider marker min
clsMarkerMax	data-cls-marker-max	Additional class for slider marker max
clsHint	data-cls-hint	Additional class for slider hint
clsHintMin	data-cls-hint-min	Additional class for slider hint min
clsHintMax	data-cls-hint-max	Additional class for slider hint max
clsMinMax	data-cls-min-max	Additional class for min-max info block
clsMin	data-cls-min	Additional class for min value for info block
clsMax	data-cls-max	Additional class for max value for info block

0

20

80

100

```
<input data-role="doubleslider"
  data-value-min="20"
  data-value-max="80"
  data-hint-always="true"
  data-hint-position-min="bottom"
  data-hint-position-max="top"
  data-show-min-max="true"
  data-cls-backside="bg-dark"
  data-cls-marker="bg-blue border-50 custom-marker"
  data-cls-hint="bg-cyan custom-marker shadow-2"
  data-cls-complete="bg-red"
  data-cls-min-max="text-bold">
```

Secret classes :)

```
<input data-role="doubleslider" class="thin">
<input data-role="doubleslider" class="ultra-thin">
<input data-role="doubleslider" class="ultra-thin cycle-marker">
```