

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
- NavView
- 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

Charms

The Charms bars is a systems toolbars that houses for important services. Create additional control area with charm component.

About charms

The Charms bars is a systems toolbars that houses for important services. The Charms bars is a vertical or horizontal toolbar, found on the: `right-side` , `left-side` , `top-side` or `bottom-side` of the screen.

To create `charms` add attribute `data-role="charms"` to block element. and setup charms position with attribute `data-position="right|left|top|bottom"` .

```
<div data-role="charms" data-position="right"></div>
```

Default charms

Position

You can setup position for `charms` . To set `position` add attribute `data-position` with one of values: `right` , `left` , `top` or `bottom` .

```
<div data-role="charms" data-position="top"><div>top</div></div>
<div data-role="charms" data-position="bottom"><div>bottom</div></div>
<div data-role="charms" data-position="left"><div>left</div></div>
<div data-role="charms" data-position="right"><div>right</div></div>
```

Top

Bottom

Left

Right

Opacity

To set `opacity` for charms use attribute `data-opacity` . The value can be between `0` and `1` .

```
<div data-role="charms" data-opacity=".5"></div>
```

opacity

Events

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

Event	Data-*	Desc
onCharmsCreate(el)	<code>data-on-charms-create</code>	Fired when charms was created
onOpen(el)	<code>data-on-open</code>	Fired when charms open
onClose(el)	<code>data-on-close</code>	Fired when charms close

```
<div data-role="charms"
  data-on-open="console.log('charms was opened')"
  data-on-close="console.log('charms was closed')"
></div>
```

Methods

You can use charms methods ti interact with the charms component.

- **open()** - open charms
- **close()** - close charms
- **toggle()** - toggle charms
- **opacity()** - get charms opacity value
- **opacity(val)** - eet charms opacity

```
var charms = $(el).data('charms');
charms.open();
charms.close();
charms.toggle();
charms.opacity(.5);
console.log(charms.opacity());
```

Also Metro 4 also implements a special object for working with the component `Metro.charms` with same methods:

- **check(el)** - Check element is a charm
- **isOpen(el)** - Check charms is open
- **open(el)** - open specific charms
- **close(el)** - close specific charms
- **toggle(el)** - toggle specific charms
- **opacity(el, opacity)** - set opacity for specific charms

Table of contents

- Charms
- About charms
- Position
- Opacity
- Events
- Methods
- Customize

- Video player
- Audio player
- Tools
- Collapse
- Color module
- Draggable
- Dropdown
- Form validator
- Hotkeys
- Micro templates
- Ripple
- Storage
- Session storage
- Sorter
- Touch and swipe

- Utilities
- Animations
- Additional

```
<div data-role="charms" id="specific-charms"></div>  
  
<script>  
    Metro.charms.open("#specific-charms");  
</script>
```

Customize

You can use attribute `data-cls-charms="..."` to set additional classes for charms.