

Search...

Getting started

- Introduction
- Download
- Contents
- Browsers & devices
- Media breakpoints

Define Events

- DOM manipulation
- i18n
- Including
- Configuring
- Init components
- Components API
- Global setup
- ReactJS
- Angular
- VueJS

Components

Utilities

Animations

Additional

Events

Events system in Metro 4 Components Library.

About

Many components in Metro 4 generate events during their work. All events are defined width attributes with prefix `data-on-*` . To define event you can use two ways:

- Set **function name** as `data-on-*` attribute value
- use **valid javascript code** for `data-on-*` attribute value

```
<input data-role="keypad" data-on-change="console.log(arguments)">
<input data-role="keypad" data-on-change="dataChange">
<input data-role="keypad" data-on-change="MyPackage.dataChange">

<script>
  function dataChange(e){
    console.log(arguments);
  }

  var MyPackage = {
    dataChange: function(e){
      console.log(arguments);
    }
  }
</script>
```

Subscribing to the events

You can subscribe to components events with m4q method `$.on(...)` and/or JS method `addEventListener(...)` .

```
<input data-role="keypad" id="keypad">

<script>
  $("#keypad").on("shuffle", function(e){
    console.log(e.detail);
  })

  /* or */

  document.getElementById("keypad").addEventListener("shuffle", function(e){
    console.log(e.detail);
  })
</script>
```

Important! If you subscribe to the events, event you must define name without **data-on-**.

```
on html
<input data-role="keypad" id="keypad" data-on-shuffle="...">

on js
document.getElementById("keypad").addEventListener("shuffle", function(e){
  console.log(e.detail);
})
```