



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

# M4Q Constructor

Constructor in m4q can create, select, import elements and generate document.ready function.

## About

Constructor in m4q can create, select, import elements and generate document.ready function.

- `$( "<div>" )` - create by `tag name`
- `$( "<div>", context )` - create in context
- `$( "<div>", {...} )` - create by tag with attributes as object
- `$( "<div>any_text_or_html</div>" )` - create by `html`
- `$( "div" )` - select by `tag name`
- `$( ".div" )` - select by `class name`
- `$( "#div" )` - select by `id`
- `$( $(...) )` - create by `m4q` as argument
- `$( jQuery(...) )` - import from jQuery
- `$( function() {} )` - Create document.ready function

## Creating elements

Created elements returns as `m4q` object. To get `HTMLElement` you must request element from m4q object by index.

### Create element by tag name

You can create an element by tag name. The created element will not append into DOM. You must insert element into DOM with one of m4q manipulation function or one of special DOM functions.

```
var div = $("<div>");

// and now add to DOM

div.appendTo($("body"));
// or
document.body.appendChild(div[0]);
```

### Create element in context

You can create element and append it to context with using second constructor parameter.

```
var context = $("body");
var div = $("<div>", context); // div added to DOM into body
```

### Create element with attributes

You can create element with required attributes. To add attributes to created element, define these in second constructor parameter.

```
var div = $("<div>", {
  id: "elementId"
  class: "my_class"
});
div.appendTo($("body"));
```

### Create elements from HTML text

You can create element(s) from html. Parameter for constructor must be a valid html string. Also you can insert created element(s) into context.

```
var div = $("<div><p>Lorem ipsum</p></div>");
div.appendTo($("body"));
```

## Selecting elements

You can selecting elements by: tag name, class name, element id and any valid css selectors.

jQuery css extensions not supported.

### Select elements by tag name

```
var divs = $("div");
var p = $("div > p");
var a = $("ul a");
```

### Select elements by class name

## Table of contents

About

- 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

```
var el = $(".my-class");
var el = $(".class-parent .class-child");
var el = $(".class.subclass");
```

Select elements by id

```
var el = $("#elem");
var el = $(".class-parent #child");
```

M4Q object as constructor argument

If you specify `m4q` object as constructor argument, M4Q return `new m4q object` with same elements.

jQuery object as constructor argument

If you specify `jQuery` object as constructor argument, M4Q imports elements from jQuery object and return `new m4q object` with same elements.

Creating document.ready function

You can specify function as constructor argument and receive document.ready function.

```
$(function () {
    ...
});

// equals to

document.addEventListener('DOMContentLoaded', function () {
    ...
});
```