

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

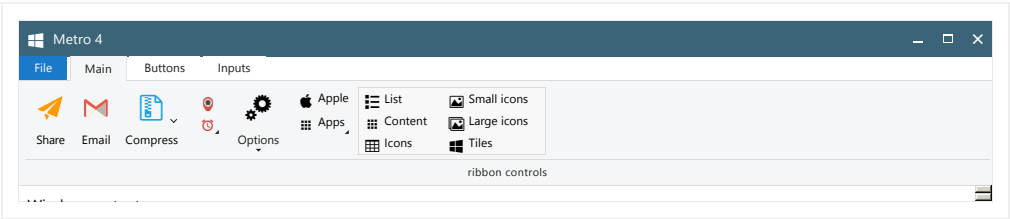
Media

# Ribbon menu

A ribbon is a command bar that organizes a program's features into a series of tabs.

## About

In computer interface design, a ribbon is a graphical control element in the form of a set of toolbars placed on several tabs. The typical structure of a ribbon includes large, tabbed toolbars, filled with graphical buttons and other graphical control elements, grouped by functionality. Such ribbons use tabs to expose different sets of controls, eliminating the need for numerous parallel toolbars. Contextual tabs are tabs that appear only when the user needs them.



## Ribbon menu structure

The Ribbon menu consists two main parts: tabs and tab content. The ribbon menu must contains role `ribbonmenu`. Tabs is a list with class `.tabs-holder` and content holder - container with class `.content-holder`.

```
<nav data-role="ribbonmenu">
  <ul class="tabs-holder">
    ...
  </ul>

  <div class="content-holder">
    ...
  </div>
</nav>
```

## Tabs

To create `tabs` you must use list with class `.tabs-holder`. Each item in tabs must be defined as `li > a`. Item can has special state `static`. To create `static` item, add class `.static` to `<li>`. In ribbon menu, static tabs have self-behavior. For disabling tab you mus use class `.disabled`.

StaticOneTwoDisabledThree

```
<nav data-role="ribbonmenu">
  <ul class="tabs-holder">
    <li class="static"><a href="#">Static</a></li>
    <li><a href="#">One</a></li>
    <li><a href="#">Two</a></li>
    <li class="disabled"><a href="#">Disabled</a></li>
    <li><a href="#">Three</a></li>
  </ul>
</nav>
```

Each tab can be linked to content section. To link use attribute `href` for `anchor`. Value for this attribute must be valid element `id`.

## Content

Content for tabs must be stored in containers with class `.section`. All section put into container with class `.content-holder`. Each section must have attribute `id` with unique value. This value will be used to link from the tab to content.

StaticOneTwoThree

Section one

```
<nav data-role="ribbonmenu">
  <ul class="tabs-holder">
    <li class="static"><a href="#">Static</a></li>
    <li><a href="#section-one">One</a></li>
    <li><a href="#section-two">Two</a></li>
    <li><a href="#section-three">Three</a></li>
  </ul>

  <div class="content-holder">
    <div class="section" id="section-one">
      <p class="p-4">Section one</p>
    </div>
    <div class="section" id="section-two">
      <p class="p-4">Section two</p>
    </div>
    <div class="section" id="section-three">
      <p class="p-4">Section three</p>
    </div>
  </div>
</nav>
```

## Section

Tabs content stored in a `section`. Section is a container with class `.section`. Each section can contains `elements`, `groups` of the elements and `group title`.

### Table of contents

- Ribbon menu
- About
- Structure
  - Tabs
  - Content
  - Section
- Ribbon elements
  - Button
  - Icon button
  - Tool button
  - Split button
  - Buttons group
  - Dropdown
  - Others...
- Events

- 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

Main

Share

Share

Share

Share

Group title

Group title

```
<nav data-role="ribbonmenu">
  <ul class="tabs-holder">
    <li><a href="#section-1-1">Main</a></li>
  </ul>

  <div class="content-holder">
    <div class="section" id="section-1-1">
      <div class="group">
        <button class="ribbon-button">
          <span class="icon">
            
          </span>
          <span class="caption">Share</span>
        </button>

        <span class="title">Group title</span>
      </div>

      <div class="group">
        <button class="ribbon-button">
          <span class="icon">
            
          </span>
          <span class="caption">Share</span>
        </button>

        <span class="title">Group title</span>
      </div>
    </div>
  </div>
</nav>
```

Ribbon elements

Metro 4 contains the number of special ribbon elements: `button` , `tool button` , `icon button` , `split button` , `buttons group` and `dropdown` .

Button

To create ribbon button add class `.ribbon-button` to `<button>` element and then add `icon` and `caption` .

Button

Share

```
<button class="ribbon-button">
  <span class="icon">
    
  </span>
  <span class="caption">Share</span>
</button>
```

Icon button

To create ribbon icon button add class `.ribbon-icon-button` to `<button>` element and then add `icon` and `caption` .

Button

Share

```
<button class="ribbon-icon-button">
  <span class="icon">
    
  </span>
  <span class="caption">Share</span>
</button>
```

Tool button

To create ribbon tool button add class `.ribbon-tool-button` to `<button>` element and then add `icon` .

Button

```
<button class="ribbon-tool-button">
  <span class="icon">
    
  </span>
</button>
```

Split button

To create ribbon split button add class `.ribbon-split-button` to `<div>` element and then add `button` , `split` and `dropdown` .

Button

Options

```

<div class="ribbon-split-button">
  <button class="ribbon-main">
    <span class="icon">
      <span class="mif-cogs"></span>
    </span>
  </button>
</div>
<span class="ribbon-split dropdown-toggle">Options</span>
<ul class="ribbon-dropdown" data-role="dropdown" data-duration="100">
  <li class="checked"><a href="#">Modification</a></li>
  <li class="checked"><a href="#">Type</a></li>
  <li class="checked"><a href="#">Size</a></li>
  <li><a href="#">Creating</a></li>
  <li><a href="#">Authors</a></li>
  <li class="checked-one"><a href="#">Tags</a></li>
  <li><a href="#">Names</a></li>
  <li class="divider"></li>
  <li><a href="#">Columns...</a></li>
</ul>
</div>

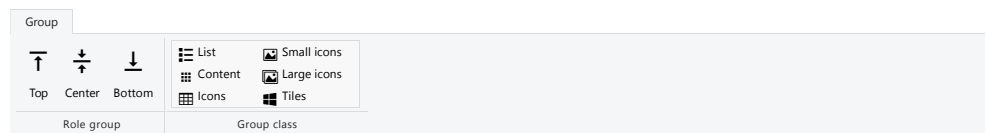
```

## Buttons group

You can create `buttons group` where `one` button has `active` state at one time. Active button has class `.active`. For this there are two ways:

Use role `data-role="buttonsGroup"` for buttons container

Use predefined class `.ribbon-toggle-group`



```

<nav data-role="ribbonmenu">
  <ul class="tabs-holder">
    <li><a href="#"section-group">Group</a></li>
  </ul>

  <div class="content-holder">
    <div class="section" id="section-group">
      <div class="group">

        <div data-role="buttonsGroup" data-cls-active="active">
          <button class="ribbon-button">
            <span class="icon">
              <span class="mif-vertical-align-top"></span>
            </span>
            <span class="caption">Top</span>
          </button>
          <button class="ribbon-button">
            <span class="icon">
              <span class="mif-vertical-align-center"></span>
            </span>
            <span class="caption">Center</span>
          </button>
          <button class="ribbon-button">
            <span class="icon">
              <span class="mif-vertical-align-bottom"></span>
            </span>
            <span class="caption">Bottom</span>
          </button>
        </div>

        <span class="title">Role group</span>
      </div>

      <div class="group">
        <div class="ribbon-toggle-group">
          <button class="ribbon-icon-button">
            <span class="icon">
              <span class="mif-list"></span>
            </span>
            <span class="caption">List</span>
          </button>

          <button class="ribbon-icon-button">
            <span class="icon">
              <span class="mif-apps"></span>
            </span>
            <span class="caption">Content</span>
          </button>

          <button class="ribbon-icon-button">
            <span class="icon">
              <span class="mif-table"></span>
            </span>
            <span class="caption">Icons</span>
          </button>

          <button class="ribbon-icon-button">
            <span class="icon">
              <span class="mif-image"></span>
            </span>
            <span class="caption">Small icons</span>
          </button>

          <button class="ribbon-icon-button">
            <span class="icon">
              <span class="mif-images"></span>
            </span>
            <span class="caption">Large icons</span>
          </button>

          <button class="ribbon-icon-button">
            <span class="icon">
              <span class="mif-windows"></span>
            </span>
            <span class="caption">Tiles</span>
          </button>
        </div>
      </div>
    </div>
  </div>

```

## Dropdown menu

Ribbon menu contains self designed dropdown menu. To create ribbon dropdown menu you must add class `ribbon-dropdown` to list. Each item can has three states: `simple` , `checked` and `checked-one` .

[illegible]

You can use ribbon dropdown menu to create button with dropped down. To create it you must use next html structure:

The screenshot shows a ribbon toolbar with a 'Dropdown' menu. The menu is open, showing 'Compress' and 'Button' options. The 'Compress' option is highlighted. The 'Button' option is also visible. The 'Compress' button is shown with a file icon.

Others...

You can put any others elements to ribbon menu. See [first example](#), input and button tabs.

## Events

When ribbon menu works, it generate any `events` . You can create `callback` to events to interact with it.

Event	Data-*	Desc
<code>onStatic(tab, el)</code>	<code>data-on-static</code>	The event occurs when the user clicks on the static tab
<code>onBeforeTab(tab, el)</code>	<code>data-on-before-tab</code>	The event occurs before tab change. If this function return false, tab not activated
<code>onTab(tab, el)</code>	<code>data-on-tab</code>	The event occurs after tab changed.
<code>onRibbonMenuCreate(el)</code>	<code>data-on-ribbon-menu-create</code>	The event occurs after menu created.

