**Options**

选项应该在初始化代码赋予

Options should be passed to the initialization code and separated by comma, e.g.:

$('.some-link').magnificPopup({

*// main options*

disableOn**:** 400,

key**:** 'some-key',

gallery**:** {

*// options for gallery*

enabled**:** **true**

},

image**:** {

*// options for image content type*

titleSrc**:** 'title'

}

});

Options for specific modules are explained in their sections of documentation (e.g. related to text are in [translating section](http://dimsemenov.com/plugins/magnific-popup/documentation.html#translating), related to gallery are in [gallery section](http://dimsemenov.com/plugins/magnific-popup/documentation.html#gallery). Here you can find the list of general options:

disableOn

null

If window width is less than the number in this option **lightbox will not be opened and the default behavior of the element will be triggered（触发）**. Set to 0 to disable behavior. Option works only when you initialize Magnific Popup from DOM element.

Can also accept Function as a parameter, which should return true if lightbox can be opened and falseotherwise. For example:

disableOn**:** **function**() {

**if**( $(window).width() **<** 600 ) {

**return** **false**;

}

**return** **true**;

}

key

null

“Key” option is a unique identifier of a single or a group of popups with the same structure. If you will not define it, DOM elements will be created and destroyed each time when you open and close popup.

You may (and should) set an equal key to different popups if their markup matches. By markup I mean options that change HTML structure of the popup (e.g. close icon placement and HTML code of it).

For example: you have many popups that show title, some text and button - you may use one key for all of them, so only one instance of this element is created. Same for popup that always contains image and caption.

You can delete cached templates like so:

// delete template with key "your-key"

delete $.magnificPopup.instance.popupsCache['your-key'];

// delete all templates

$.magnificPopup.instance.popupsCache = {};

midClick

false鼠标中键

If set to true lightbox is opened if the user clicked on the middle mouse button, or click with Command/Ctrl key. Option works only when you initialize Magnific Popup from DOM element.

mainClass

empty string

String that contains classes that will be added to the root element of popup wrapper and to dark overlay（覆盖）. For example "myClass", can also contain multiple classes - 'myClassOne myClassTwo'.

一个字符串that包含了classes that将要被加入到popup根元素that被popup包裹的和面向黑色覆盖层的。

一个类能包含多个类

意思就是，将要往初始化的那个popup增加新的部分

preloader

true

Preloader in Magnific Popup is used as an indicator of current status. If option enabled, it’s always present in DOM only text inside of it changes. Below you can see explanation of CSS names that are applied to container that **holds preloader and content area** depending on the state of current item: **监听器**

*/\* Content loading is in progress \*/*

**.mfp-s-loading** { }

*/\* Content successfully loaded \*/*

**.mfp-s-ready** { }

*/\* Error during loading \*/*

**.mfp-s-error** { }

For example, if you want your error message to be in red add such CSS:

**.mfp-s-error** **.mfp-preloader** {

**color:** red;

}

You can trigger change of status manually by calling instance.updateStatus('error', 'error message').

focus

empty string

String with CSS selector of an element inside popup that should be focused. Ideally it should be the first element of popup that can be focused. For example 'input' or '#login-input'. Leave empty to focus the popup itself.

closeOnContentClick

false

Close popup when user clicks on content of it. It’s recommended to enable this option when you have only image in popup.

closeOnBgClick

true

Close the popup when user clicks on the dark overlay.

closeBtnInside

true

If enabled, Magnific Popup will put close button inside content of popup, and wrapper will get class mfp-close-btn-in (which in default CSS file makes color of it change). If markup of popup item is defined element with class mfp-close it will be replaced with this button, otherwise close button will be appended directly.

showCloseBtn

true

Controls whether the close button will be displayed or not.

enableEscapeKey

true

Controls whether pressing the escape key will dismiss the active popup or not.

modal

false

When set to true, the popup will have a modal-like behavior: it won’t be possible to dismiss it by usual means (close button, escape key, or clicking in the overlay).

This is a shortcut to set closeOnContentClick, closeOnBgClick, showCloseBtn, and enableEscapeKey to false.

alignTop

false

If set to true popup is aligned to top instead of to center. (basically all this option does is adds mfp-align-topCSS class to popup which removes styles that align popup to center).

index

null

Used for gallery. Defines starting index. If popup is initialised from DOM element, this option will be ignored.

fixedContentPos 固定内容位置

auto

Popup content position. Can be "auto", true or false. If set to true - fixed position will be used, to false- absolute position based on current scroll. If set to "auto" popup will automatically disable this option when browser doesn’t support fixed position properly.

fixedBgPos 固定背景位置

auto

Same as an option above, but it defines position property of the dark transluscent overlay. If set to false - huge tall overlay will be generated that equals height of window to emulate fixed position. It’s recommended to set this option to true if you animate this dark overlay and content is most likely will not be zoomed, as size of it will be much smaller.

overflowY

auto

Defines scrollbar of the popup, works as overflow-y CSS property - any [CSS acceptable value](https://developer.mozilla.org/en-US/docs/CSS/overflow-y) is allowed (e.g. auto, scroll, hidden). Option is applied only when fixed position is enabled.当位置固定时，才能使用scrollBar

There is no option overflowX, but you may easily emulate it just via CSS.

没有横向scrollBar，但你可以自己在CSS定义

removalDelay

0

Delay before popup is removed from DOM. Used for the [animation](http://dimsemenov.com/plugins/magnific-popup/documentation.html#animation).

closeMarkup

<button title="%title%" type="button" class="mfp-close">&#215;</button>

Markup of close button. %title% will be replaced with option tClose.

prependTo

document.body

The DOM element to which popup will be added. Useful when you’re using ASP.NET where popup should be inside form. Option available since 2013/12/04.

autoFocusLast

true

If set to true last focused element before popup showup will be focused after popup close. Option available since 2015/12/16.