

# JavaScript Popup Generator



## Disclaimer

This SOFTWARE PRODUCT is provided by El Condor "as is" and "with all faults." El Condor makes no representations or warranties of any kind concerning the safety, suitability, lack of viruses, inaccuracies, typographical errors, or other harmful components of this SOFTWARE PRODUCT. There are inherent dangers in the use of any software, and you are solely responsible for determining whether this SOFTWARE PRODUCT is compatible with your equipment and other software installed on your equipment. You are also solely responsible for the protection of your equipment and backup of your data, and El Condor will not be liable for any damages you may suffer in connection with using, modifying, or distributing this SOFTWARE PRODUCT.

You can use this SOFTWARE PRODUCT freely, if you would you can credit me in program comment:

El Condor – CONDOR INFORMATIQUE – Turin

Comments, suggestions and criticisms are welcomed: mail to [rossati@libero.it](mailto:rossati@libero.it)

## Conventions

Commands syntax, instructions in programming language and examples are with font COURIER NEW. The optional parties of syntactic explanation are contained between [square parentheses], alternatives are separated by | and the variable parties are in *italics*.

## Contents table

1	PopUp generator.....	1
1.1	Using the PopUp generator.....	1
1.2	Compatibility.....	1
1.3	Parameters description.....	1
1.4	How works.....	1
1.4.1	Popup creation.....	1
1.4.2	Popup closing.....	1
1.5	Parameters.....	1
1.5.1	Cancel.....	1
1.5.2	Class.....	1
1.5.3	Content.....	1
1.5.4	Fade.....	2
1.5.5	Id.....	2
1.5.6	Left and Top.....	2
1.5.7	Width and Height.....	2
1.6	Events.....	2
2	Examples.....	2
2.1	Center.....	2
2.2	Cancel by Escape key.....	2
2.3	Fading.....	2
2.4	Show random image.....	2
3	History.....	3

## 1 PopUp generator

This is a documentation on the JavaScript program `popUpGen.js` for generate popups.

### 1.1 Using the PopUp generator

The popup is created by calling the function:

```
popUp(parameters)
```

where *parameters* is an object that contains the parameters necessary to personalize the popup, see some examples at paragraph 2 Examples.

### 1.2 Compatibility

Needs HTML5 for access the size of images.

### 1.3 Parameters description

The parameters are indifferent to the case.

Name	Default value	Note
cancel	click	Cancel by Click or key (see <a href="#">here</a> for values), or number of milliseconds
class	empty	
content	empty	Is the content that must be shown
fade	empty	From 0 to two numeric values, the first one is fade in, the second one the fade out value in milliseconds
height	-1	On default the height is automatically determined
id	empty	
left	-1	The default value forces horizontal centering
top	-1	The default value forces vertical centering
style	empty	
width	-1	On default the width is automatically determined

### 1.4 How works

#### 1.4.1 Popup creation

The popup is inserted in a `div` tag newly created; if the *id* exists the `div` becomes his child otherwise is inserted at the beginning of the page:

```
(document.body.insertAdjacentElement('afterbegin', link))
```

#### 1.4.2 Popup closing

The `div` created is removed.

### 1.5 Parameters

#### 1.5.1 Cancel

`click|no|key|number`

This indicates how to close the popup, it can contains a *key* (see [here](#) for values) or `click` or `no` or a numeric value that is the number of milliseconds of the popup life. `no` means that the popup remains on the page. `click` or `no` are indifferent to case.

Some possible *keys* are: Delete, Enter, Escape.

In case of *number* the duration is inclusive of any *fade* duration.

#### 1.5.2 Class

Adds a `class` in the `div` created for receive the popup.

#### 1.5.3 Content

Is the popup content that is inserted on the `div` created.

### 1.5.4 Fade

Can contains up to two numbers: the first one is fade in time, the second one the fade out time; the time values are in milliseconds; if only fade out is requested the fade in must be present with value 0: "fade": "0 number".

### 1.5.5 Id

is the `Id` of the element after which the `content` of the popup will be inserted.

### 1.5.6 Left and Top

These parameters are taken into account if a container `Id` is not indicated or this doesn't exist. If the value is `-1` (the default) the popup is centered than the width (`left = -1`) or height (`top = -1`).

### 1.5.7 Width and Height

They determine the size of the popup, it makes sense for popups with images. A value `-1` doesn't change the relative dimension.

Setting `width` and `height` at the same time can possibly distort image, vice versa, setting one of the parameters maintains the proportions of a possible image.

## 1.6 Events

A popup is created by default with `click` event on the popup or with the event `keydown` unless the case of popup splash that disappears after a time indicated by `cancel=numberOfMilliseconds`.

## 2 Examples

### 2.1 Center

Only the `content` must be provided.

### 2.2 Cancel by Escape key

In addition to the `content`, the `cancel` parameter must be set.

```
popup({'content': '...', 'cancel':'Escape', ...})
```

### 2.3 Fading

The `fade` can affect both the display and the disappearance of the popup.

```
popup({'content': '...', 'fade':'2000',...}) // fade in two seconds
popup({'content': '...', 'fade':'0 2000',...}) // fade out two seconds
```

### 2.4 Show random image

```
Var parms = {...}
parms["content"] = "<img src='getImage.php?R="+Date.now()+"'>";
```

Below an example of sending a random image by a PHP script on server. The `?R="+Date.now()` is to avoid the image caching.

```
<?php
function randomImg($folder) {
    $aImages = glob($folder . '/*.{jpg,jpeg,png,gif}', GLOB_BRACE);
    shuffle($aImages);
    return array_pop($aImages);
}
$fileImage = randomImg("images");
$aMimeTypes = array('jpeg' => 'image/jpeg',
                    'jpg' => 'image/jpeg',
                    'png' => 'image/png',
                    'gif' => 'image/gif',);
$path_parts = pathinfo($fileImage);
```

```
header("Content-type: ".$aMimeTypes[$path_parts['extension']);  
readfile("$fileImage");  
exit(0);  
?>
```

### 3 History

0.1.0 30 March 2020 First version