Multimodalbox

OVERVIEW

Multimodalbox is a lightweight customizable lightbox plugin that supports several instances in the same template.

Multimodalbox uses jquery ui dialogs and accepts settings in a very similar way to Colorbox, but the vast majority of the settings in Colorbox are not implemented in multimodalbox. Let me know if you're missing anything.

Bug in Windows

Don't use a name for the variable holding the multimodalbox equal to the id of any element in the DOM

Wrong

```
<div id="hidden_modal"></div>
...
var hidden_modal;
```

This causes a bug in Windows.

The explanation is that If a variable is undefined, Windows try to get a DOM element with the same name. Therefore, this typical implementation fails:

```
if (!hidden_modal){ //Instead of being evaluated to "undefined" Windows retrieves
the HTML element
        hidden_modal = new multimodalbox({.....});  //Windows never enters here
}else{
         hidden_modal.open() // ERROR!! The div with id "hidden_modal" doesn't have
a method called open (how dare it!!)
}
```

To sum up, don't use a name for the multimodalbox variable equal to any element in the DOM.

SETTINGS

Setting	Type	Default	Description
element	str	false	Is the jquery identifier of the element that will hold the dialog. If the element doesn't exist multimodalbox will append a div to the body with id = "element". It is a required setting.
data	dict	false	dictionary to send in an AJAX call. When false, multimodalbox uses GET calls. If "data" is not false, multimodalbox will use a POST call.
inline	boolean	false	If true, content from the current document can be displayed using the href property to indicate a jQuery selector, or jQuery object.
iframe	boolean	false	If true displays the url in href as an iframe.
photo	boolean	false	If true displays the url in href as the src field in an img tag
html	str	false	If true displays its content inside an multimodalbox (can display HTML or text).
uiblocker	boolean	false	if true displays a loading gif and blocks the UI untill the multimodalbox is closed.
href	str	false	Depending on one of the previous options it can be the url for an ajax request, the jquery selector for an inline element, the url for an iframe or the url for an image
width	str	"auto"	Set a fixed total width. This includes borders and buttons. It accepts the formats <number> + "px" or <number> + "%". Examples: "250px", "85%" It is not a required setting but it is strongly recommended to use it always.
height	str	"auto"	Similar to width
maxWidth	str	"90%"	
maxHeight	str	"90%"	
onOpen	function	false	Callback that fires right before multimodalbox begins to open,
onLoad	function	false	Callback that fires right after multimodalbox is opened and ready to use.
onCleanUp	function	false	Callback that fires right after the jquery ui dialog is destroyed
onClose	function	false	Callback that fires right after the multimodalbox in closed
dialog	dictionary	-	Dictionary with default values for jquery ui dialog. You can edit this values using this setting
open	boolean	true	If true the multimodalbox is opened right after it is created.

METHODS

Since we can have several instances of multimodalbox we shouldn't need to resize or do dramatic updates of a lightbox content, and in most of the cases we'll only need to use: multimodalbox(), open() and close().

Multimodalbox copies the element
Beware that multimodalbox creates a copy of the element and place it inside a jquery ui dialog

Constructor

The constructor method. Receives a dictionary with settings and creates and opens an multimodalbox.

create a new multimodalbox object

```
var new_popup = new multimodalbox({element:"#popup_container", html:"<strong>All the
glory to the hypnotoad!!</strong>"});
```

Open method

Opens and already existing multimodalbox.

```
new_popup.open();
```

Close method

Closes an existing multimodalbox

```
new_popup.close();
```

Update method

Update the settings of an existing multimodalbox. Remember that you can use several multimodalbox at once! don't try to do weird stuff like trying to display an iframe in the multimodalbox you were using to do AJAX calls or trying to update the "element" setting of an existing multimodalbox.

Would it explode? I don't think so, but it's a recipe for disaster.

Update method

```
new_popup.update({html:"I don't need HTML tags!", width: "600px"});
```

Remove method

Removes an multimodalbox from the document, destroys the dialog and unbinds the events.

```
new_popup.remove();
```

Resize method

Resize an existing multimodalbox to a new width and height. Optionally you can add the duration of the animation in miliseconds (default value is 400ms)

Resize method

```
// resize(width, height, mseconds)
new_popup.resize("60%", "300px", 500);
```

EVENTS

To recognize which multimodalbox is producing the event we use the value of "element" as a prefix.

```
image_overlay = new multimodalbox({element: "#image_div", photo: true, href:
"../img/moustachio.png", width: "500px"});
```

In this case:

prefix = "image_div"   (without '#')

Event name	Description	Equivalent callback
prefix + "_ready"	Gets fired once multimodalbox it's opened and ready. It only gets fired when creating or updating an multimodalbox	onLoad
prefix +	Gets fired everytime the multimodalbox is opened	onOpen

"_open"		
prefix + "_cleanup"	Gets fired once the multimodalbox is destroyed	onCleanUp
prefix + "_closed"	Gets fired everytime the multimodalbox is closed.	onClose

The event "_ready" is really useful to start the listeners inside the advent box.

```
$("#image_overlay").bind("image_overlay_ready", function(){

        $("#cancel_button").click(function(){
                image_overlay.close();
        });

});
```

EXAMPLES

Use multimodalbox to display an inline document:

```
var efin_modal_v;   //Remember: do not use the same name for the variable and the
multimodalbox 'element'.


if (!efin_modal_v){
        //if the multimodalbox doesn't exist create it
        efin_modal_v = new multimodalbox({element:"#efin_modal", inline" true,
href:"#efin_modal", onClose: function(){ reload_window();} });
}else{
        //if it already exists we only have to open it.
        efin_modal_v.open()
}


$("#efin_modal").one("efin_modal_ready", function(){

        $("#ok_button").click(function(){
                approve_efin();
        });

        $("#cancel_button").click(function(){
                efin_modal_v.close();
        });

        initForm();
        //...

});
```

Use multimodalbox to display content via AJAX:

```
var document_overlay;

if (!document_overlay){
        //if the multimodalbox doesn't exist create it
        document_overlay = new multimodalbox({
                element: "#review_overlay",
        width: "1100px",
        maxHeight: "90%",
        href: "/agent/review_document_overlay",
        data: {"a_id": "34de46084ab3"},
        onClose: function() {
        reload_window();
                },
                onLoad: function() {
                        listeners();
                }
        });
}else{
        //if it already exists update it.
        document_overlay.update({data: {"a_id": "aa25647df34d"}})
}
```

Use multimodalbox as a UI blocker:

```
var loading = new multimodalbox({uiblocker:true}); //no need to specify element or any
other option

// to close it..

loading.close();  // surprise!!
```