

Documentation

The Basics

The following is a very basic demonstration of how to display an alert.

```
1  /* very basic demonstratin of jqAlert; showing a basic alert styled with your current jQuery UI Theme
2  $.alert('Your message goes here.');
```

However, in most cases you are probably going to want more interaction with the dialog that is displayed to your users. With each of the dialogs you get access to the following common properties.

| | |
|--------------------|---|
| title | Defines the title displayed on the dialogs title bar. |
| icon | Defines the icon displayed with the message. Short-hand version of the jQuery glyphs (subtract ui-icon-). |
| customIcon | Property that allows you to define a custom icon created in your CSS. The icon will have to be url encoded as a class in CSS. |
| show | The property that allows you to define the animation for how the dialog is displayed. |
| hide | The property that allows you to define the animation for how the dialog is closed. |
| timer | Define a timeout for how long the dialog will be displayed. Supplying a value of <code>-1</code> will disable the timer. |
| allowEscape | Property that allows you to set whether a dialog can be closed by hitting the escape key or not. |

There is also a common property for the buttons that gets displayed with your dialog.

| | |
|----------------|---|
| buttons | The property that defines the array of buttons to be displayed with the dialog. |
|----------------|---|

Button properties can include the following:

| | |
|-----------------|--|
| title | The property that defines the title text displayed on the button face. |
| callback | A function property to be used as a callback for when the button is pressed. |
| css | This property allows you to associate a custom css class to button. |

Finally you have the following callback notifications for dealing with a dialog:

| | |
|------------------|--|
| onTimeout | In the event that a timer value was given this callback is used to notify that the timeout has been reached. |
| onClose | A callback notification that when supplied will notify when the dialog has been closed. |

It sounds like a lot for a simple dialog but the goal is to serve those that want it all at times. However, strides have been made to still keep it fairly simple.

So bringing it all together we can create something like the following:

```
1  /* bring it all together */
2  $.alert('jqAlert is easy to use with alerts and its where I got my start.', {
3      title: 'Alert Message',
4      icon: '',
5      customIcon: 'icon-asterisk',
6      buttons: [
7          {
8              title: 'Ok',
9              callback: function() { $(this).dialog("close"); },
10             css: 'greenButton'
11          }
12      ]
13  });
```

Display

Alert Dialogs

Aside from the default properties alert dialogs can be loaded with the following:

You can read the [History](#) section for details as to where these properties came from.

- exception** The property that defines the name or type of exception message.
- stack** The property that defines the stack for the associated exception message.

If no buttons are supplied the alert dialog displays the following by default:

Ok Ok button for user input.

Example:

```
1  /* bring it all together */
2  $.alert('Here is a very basic alert message.', {
3      title: 'Basic Alert',
4      icon: 'alert',
5      exception: 'Exception Name',
6      stack: 'System stack'
7  });
8
```

Display

Taking it even further we can do the following which uses most of the properties available to an alert dialog.

```
1  /* bring it all together and make something more detailed */
2  $.alert('jqAlert is easy to use with alerts and its where I got my start.', {
3      title: 'Alert Title Message',
4      icon: '',
5      customIcon: 'icon-asterisk',
6      exception: 'Exception',
7      stack: 'Stack Trace',
8      timer: 5000,
9      onTimeout: function() { $.alert('timed out'); },
10     onClose: function() { $.alert('closed'); },
11     buttons: [
12         {
13             title: 'Press Me',
14             callback: function() { $(this).dialog("close"); }
15         }
16     ]
17 });
18
```

Display

Confirmation Dialogs

The confirmation dialog supports all of the common properties and if no buttons are supplied it displays the following:

- Yes** Yes button for user input.
- No** No button for user input.
- Cancel** Cancel button for user input.

Example:

```
1  /* bring it all together */
2  $.confirm('So from this dialog do you notice the confirmation buttons?', {
3      title: 'Confirmation Prompt'
4  });
5
```

Display

Let's take it even further and show how we can use all of the properties available.

```
1  /* bring it all together and make something more detailed */
2  $.confirm('jqAlert is easy to use when you want to get a confirmation.', {
3      title: 'Confirm Message Title',
4      icon: '',
5      customIcon: 'icon-asterisk',
6      timer: 5000,
7      onTimeout: function() { },
8      onClose: function() { },
9  });
```

```

9     buttons:[
10         {
11             title:'Yes',
12             callback:function() { $(this).dialog("close");},
13             css:'greenButton'
14         },
15         {
16             title:'No',
17             callback:function() { $(this).dialog("close");},
18             css:'redButton'
19         }
20     ]
21 });
22

```

Display

Prompt Dialogs

Aside from the default properties prompt dialogs can also be configured with the following:

defaultResult The property that defines the default message a user can supply.

If no buttons are supplied the alert dialog displays the following by default:

Ok Ok button for user input.
Cancel Cancel button for user input.

Example:

```

1  /* bring it all together */
2  $.prompt('So from this dialog you can prompt the user for information.', {
3      title:'Basic Prompt',
4      defaultResult:'Default user message'
5  });
6

```

Display

Let's take it even further and show how we can use all of the properties available.

```

1  /* bring it all together and make something more detailed */
2  $.prompt('jqAlert is easy to use with prompt dialogs that get information from the user.', {
3      title:'Prompt Message Title',
4      icon:'',
5      customIcon:'icon-asterisk',
6      defaultResult:'Some default response',
7      timer:10000,
8      onTimeout:function() { $.alert('timed out');},
9      onClose: function() { },
10     buttons:[
11         {
12             title:'Ok',
13             callback:function() {
14                 $.alert($(this).find('#result').val());
15                 $(this).dialog("close");
16             }
17         },
18         {
19             title:'Cancel',
20             callback:function() { $(this).dialog("close");}
21         },
22         {
23             title:'Help',
24             callback:function() { $(this).dialog("close");}
25         }
26     ]
27 });
28

```

Display

Info Dialogs

The inform dialog supports all of the default properties and callback notifications. The only exception that the inform dialog has over the dialogs is that it does not support buttons.

Example

```
1 // an inform dialog with an 8 second timeout
2 $.inform('jqAlert is easy to use when you want an timed info message.', {
3   title: 'Inform Title',
4   timer: 8000,
5   icon: '',
6   customIcon: 'icon-asterisk',
7   allowEscape: false,
8   onTimeout: function() { $.alert('timed out');}
9 });
10
```

Display

