

Dialoghi predefiniti

Esistono tutta una serie di function predefinite per aprire dei dialog dojo standard per i casi più comuni. In generale se ne può settare il titolo ed un messaggio nel contenuto. E' possibile inoltre eseguire callback alla pressione dei vari button presenti nelle varie tipologie predefinite di dialog.

- [Show Ok Dialog](#)
- [Show Yes No Dialog](#)
- [Show Yes Cancel Dialog](#)
- [Show Yes No Cancel Dialog](#)
- [Show Multiple Buttons Dialog](#)
- [Show Error Dialog](#)

Show Ok Dialog

Apri un dialog dojo con un titolo, che mostra un messaggio e con il solo tasto (localizzato) 'Ok'. Al click su 'Ok' il dialog viene chiuso. Opzionalmente si può eseguire un'azione al click sul button 'Ok'.

Parametri:

- **title** String, optional
- **message** String, optional
- **okCallback** function, optional. Funzione eseguita al click sul button 'Ok'

Parametri @Deprecated:

- **yesCallback** function, optional. @Deprecated ⇒ okCallback (from v4.4.0)

Esempio:

```
var title = 'Visualized Title'; //optional
var message = 'Visualized Message'; //optional
var okCallback = function(){ //optional
    alert('Ok Button Clicked');
};
var params = { title: title, message: message, okCallback: okCallback};
showOkDialog(params);
```

Show Yes No Dialog

Apri un dialog dojo con un titolo, che mostra un messaggio e con i tasti (localizzati) 'Yes' e 'No'. Al click su tutti i button il dialog viene chiuso. Opzionalmente si può eseguire un'azione al click su ognuno dei button.

Parametri:

- **title** String, optional

- **message** String, optional
- **yesCallback** function, optional. Funzione eseguita al click sul button 'Yes'
- **noCallback** function, optional. Funzione eseguita al click sul button 'No'

Esempio:

```
var title = 'Visualized Title'; //optional
var message = 'Visualized Message'; //optional
var yesCallback = function(){
    alert('Yes Button Clicked');
};
var noCallback = function(){
    alert('No Button Clicked');
};
var params = { title: title, message: message, yesCallback: yesCallback,
noCallback: noCallback};
showYesNoDialog(params);
```

Show Yes Cancel Dialog

Apri un dialog dojo con un titolo, che mostra un messaggio e con i tasti (localizzati) 'Yes' e 'Cancel'. Al click su tutti i button il dialog viene chiuso. Opzionalmente si può eseguire un'azione al click su ognuno dei button.

Parametri:

- **title** String, optional
- **message** String, optional
- **yesCallback** function, optional. Funzione eseguita al click sul button 'Yes'
- **cancelCallback** function, optional. Funzione eseguita al click sul button 'Cancel'

Esempio:

```
var title = 'Visualized Title'; //optional
var message = 'Visualized Message'; //optional
var yesCallback = function(){
    alert('Yes Button Clicked');
};
var cancelCallback = function(){
    alert('Cancel Button Clicked');
};
var params = { title: title, message: message, yesCallback: yesCallback,
cancelCallback: cancelCallback};
showYesCancelDialog(params);
```

Show Yes No Cancel Dialog

Apri un dialog dojo con un titolo, che mostra un messaggio e con i tasti (localizzati) 'Yes' e 'Cancel'. Al click su tutti i button il dialog viene chiuso. Opzionalmente si può eseguire un'azione al click su ognuno dei button.

Parametri:

- **title** String, optional
- **message** String, optional
- **yesCallback** function, optional. Funzione eseguita al click sul button 'Yes'
- **noCallback** function, optional. Funzione eseguita al click sul button 'No'
- **cancelCallback** function, optional. Funzione eseguita al click sul button 'Cancel'

Esempio:

```
var title = 'Visualized Title'; //optional
var message = 'Visualized Message'; //optional
var yesCallback = function(){
    alert('Yes Button Clicked');
};
var noCallback = function(){
    alert('No Button Clicked');
};
var cancelCallback = function(){
    alert('Cancel Button Clicked');
};
var params = { title: title, message: message, yesCallback: yesCallback,
noCallback: noCallback, cancelCallback: cancelCallback};
showYesCancelDialog(params);
```

Show Multiple Buttons Dialog

Apri un dialog dojo con un titolo, che mostra un messaggio e con il set di button come da parametro in ingresso. Al click, opzionalmente, si può far eseguire una specifica azione.

Parametri:

- **title** String, optional
- **message** String, optional
- **buttons** Object[], optional. E' un array di oggetti javascript, ognuno contenente le informazioni basilari per la definizione di un tasto: etichetta, icona e azione da eseguire al click. Il comportamento eseguito al click è totalmente configurabile quindi, se necessario, si deve specificare l'istruzione per chiudere il dialog nel momento voluto. Template elemento di buttons:

```
{
    label: positiveResponseLabel,
    iconClass: 'iconDone',
```

```

    action: function(dialog, event){
        dialog.hide();

        //add code here
    }
}

```

Esempio:

```

var title = confirmRequestLabel;
var message = deleteConfirmQuestionMessage;
var buttons = [
    {
        label: positiveResponseLabel,
        iconClass: 'iconDone',
        action: function(dialog, event){
            dialog.hide();

            //add code
        },
    },
    {
        label: cancelLabel,
        iconClass: 'iconCancel',
        action: function(dialog, event){
            dialog.hide();
        }
    }
];

var params = { title: title, message: message, buttons: buttons};
showMultipleButtonsDialog(params);

```

Show Error Dialog

Apri un dialog dojo di errore con un titolo predefinito localizzato 'Errore', che visualizza il messaggio passato come parametro accanto ad una icona di errore e una toolbar in basso con un button 'Ok'. Al click sul button si chiude il dialog. Opzionalmente si può eseguire una callback al click del tasto.

Parametri:

- **message** String, optional
- **options** Object, optional.
 - **escapeHTML** Boolean, optional, default false
 - **okButtonClickCallback** function, optional, default null

Esempi:

```
var message = 'I am an error message';
showErrorDialog(message);

var message = 'I am a <h1>big</h1> error message';
var options = {
  escapeHTML: true,
  okButtonClickCallback: function(){
    //TODO add code here
  }
};
showErrorDialog(message, options);
```

From:

<https://wiki.geowebframework.com/> - **GeowebFramework**

Permanent link:

https://wiki.geowebframework.com/doku.php?id=custom:api_js_open_predefined_dialog

Last update: **2019/11/06 09:42**

