

Phalcon-jquery : JQuery

Principales fonctionnalités de la librairie Phalcon-JQuery (hors JQuery UI et Twitter Bootstrap)

-- Association d'évènements

-- Evénements connus

-- Exemple

Click sur un bouton d'id **#btn** :

```
public function clickAction(){
    $this->click("#btn","console.log('click sur #btn');");
    $this->query->compile($this->view);
}
```

Vue associée :

```
<input type="button" value="Test click" id="btn">
{{script_foot}}
```

-- Utilisation du paramètre event du callback

Le code passé au callback peut utiliser le paramètre **event** passé par JQuery, et **\$(this)**, la cible de l'évènement (target) :

Exemple : appui sur une touche, récupération du code de la touche frappée et de l'id de l'élément ayant généré l'évènement.

```
public function clickAction(){
    $this->jquery->keydown("#btn","console.log('code de touche appuyée : ' +
event.which + ' sur #' + $(this).attr('id'));");
    $this->jquery->compile($this->view);
}
```

Exemple : affichage/masquage d'un élément sur changement de la valeur d'une checkbox :

```
public function checkedAction(){
    $this->jquery->change("#ck",$this->jquery->toggle("#fs",$(this).checked()));
    $this->jquery->compile($this->view);
}
```

Vue associée :

```
<input type="checkbox" id="ck" checked>
<fieldset id="fs">
<legend>Elément à afficher/masquer</legend>
</fieldset>
{{script_foot}}
```

-- Liste des méthodes JQuery liées aux évènements

Méthode	Evènement
blur(selector,callback)	Sur perte du focus par l'élément désigné par selector
change(selector,callback)	Sur changement de valeur d'un input , select ou textarea (déclenché généralement sur perte focus)
click(selector,callback)	Sur click de l'élément désigné par selector
dblclick(selector,callback)	Sur double-click de l'élément désigné par selector
error(selector,callback)	Sur erreur de chargement sur selector
focus(selector,callback)	Sur réception du focus par l'élément désigné par selector
hover(selector,callback)	Sur déplacement de la souris au dessus de l'élément désigné par selector
keydown(selector,callback)	Sur touche clavier enfoncée lorsque l'élément désigné par selector a le focus
keypress(selector,callback)	Sur touche clavier enfoncée lorsque l'élément désigné par selector a le focus
keyup(selector,callback)	Sur touche clavier relâchée lorsque l'élément désigné par selector a le focus
load(selector,callback)	Sur chargement de l'élément désigné par selector
mousedown(selector,callback)	Sur bouton souris enfoncé lorsque l'élément désigné par selector
mouseout(selector,callback)	Sur sortie de la souris de l'élément désigné par selector
mouseover(selector,callback)	Sur passage souris au dessus de l'élément désigné par selector
mouseup(selector,callback)	Sur bouton souris relâché lorsque l'élément désigné par selector
ready(callback)	Sur chargement du document terminé
scroll(selector,callback)	Sur défilement dans/sur l'élément désigné par selector
trigger(selector,event)	Déclenche l'évènement event sur l'élément désigné par selector
unload(selector,callback)	Sur déchargement de l'élément désigné par selector

-- Evènements non listés

Pour les évènements inconnus de JQuery ou ne disposant pas de méthode associée dans la librairie phalcon-jquery, on utilise les méthodes :

- `execAndBindTo(selector,event,callback)` permettant d'exécuter le **callback** sur le déclenchement de **event** sur l'élément désigné par **selector**.
- `dojQueryAndBindTo($element,$event,$elementToModify,$jqueryCall,$param="", $function="")` permettant d'exécuter la méthode jquery **jqueryCall** sur **elementToModify**, en lui passant **param** sur le déclenchement de **event** sur l'élément désigné par **element**.

-- Requêtes Ajax

-- Requêtes immédiates

Il s'agit des requêtes dont l'exécution n'est pas différée (non associée à un évènement par exemple).

--GET

Exemple : Chargement de **exemple/getResponse** dans la div **#divResponse**

```
public function getAction(){
    $this->jquery->get("exemple/getResponse", "#divResponse");
    echo "<div id='divResponse'>Chargement...</div>";
    echo $this->jquery->compile();
}
public function getResponseAction(){
    echo "Chargement terminé";
    $this->view->disable();
}
```

-- Manipulation du DOM

-- Utilitaires

From:

<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:

<http://slamwiki2.kobject.net/slam4/php/phalcon/jquery/usage?rev=1426637410>

Last update: **2019/08/31 14:42**

