

Bootstrap 4 : fenêtre modale (Modal)

Une fenêtre modale apparaît au-dessus de la page et est en général utilisée comme une boîte de dialogue avec l'utilisateur. Elle comporte généralement un formulaire. Nous allons voir dans ce chapitre comment créer et paramétrer une fenêtre modale.

Le principe

Ce plugin nécessite 2 éléments :

- un déclencheur avec l'attribut **data-toggle= »modal »**,
- un contenu avec la classe **modal**.

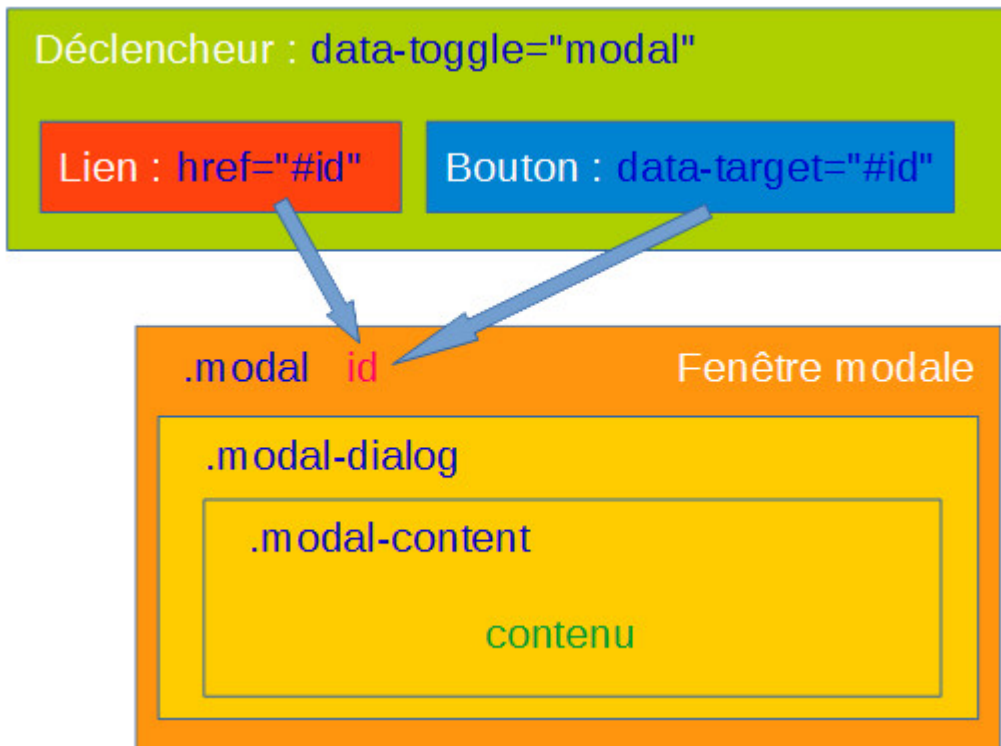
Mais comme il n'y a pas de contenant il faut établir un lien entre le déclencheur et le contenu. Cela se fait en attribuant au contenu un identifiant qui sera référencé par le déclencheur avec deux cas :

- **un lien** : la référence se fait avec un attribut **href**,
- **un bouton** : la référence se fait avec un attribut **data-target**.

Mais il nous faut aussi des classes pour styliser la fenêtre modale avec une largeur, un positionnement, un fond, une bordure... Pour réaliser tout ça on va utiliser deux classes :

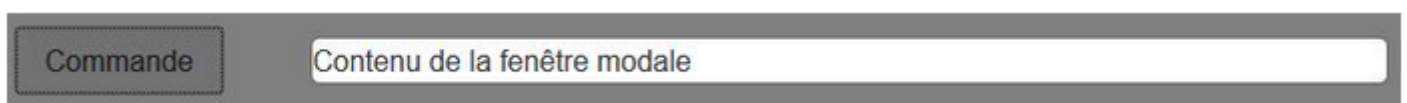
- **modal-dialog** : fixe la position, la largeur et les marges,
- **modal-content** : fixe le fond et les bordures.

Ce qu'on peut schématiser ainsi :



Donc dans sa version la plus épurée on a ce code avec un bouton comme déclencheur :

```
<button type="button" data-toggle="modal" data-target="#infos"
class="btn btn-secondary">Commande</button>
<div class="modal" id="infos">
  <div class="modal-dialog">
    <div class="modal-content">
      Contenu de la fenêtre modale
    </div>
  </div>
</div>
```



Entête, corps et pied

Maintenant que nous avons vu le principe de fonctionnement prenons un exemple avec :

- une entête avec la classe **modal-header**, avec possibilité d'un titre avec la classe **modal-title**,
- un corps avec la classe **modal-body**,

- un pied de page avec la classe **modal-footer**.

```
<button type="button" data-toggle="modal" data-target="#infos"
class="btn btn-primary">Informations</button>
<div class="modal" id="infos">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">Plus d'informations</h4>
      </div>
      <div class="modal-body">
        Le Tigre (Panthera tigris) est un mammifère carnivore de
la famille des félidés...
      </div>
      <div class="modal-footer">
        <em>Informations sous réserve</em>
      </div>
    </div>
  </div>
</div>
</div>
```

[Tester en ligne](#)



Ce

qui peut se schématiser ainsi :



Commande de fermeture

Dans les exemples vus ci-dessus pour fermer la fenêtre modale il faut cliquer n'importe où sur la page (en dehors de la fenêtre). Ce n'est pas vraiment très pratique. Il serait bien d'avoir un bouton dans la fenêtre pour assurer sa fermeture. Pour le réaliser on a l'attribut **data-dismiss= »modal »**. Classiquement on prévoit un petit bouton en forme de croix dans l'entête ou/et un bouton en pied de fenêtre.

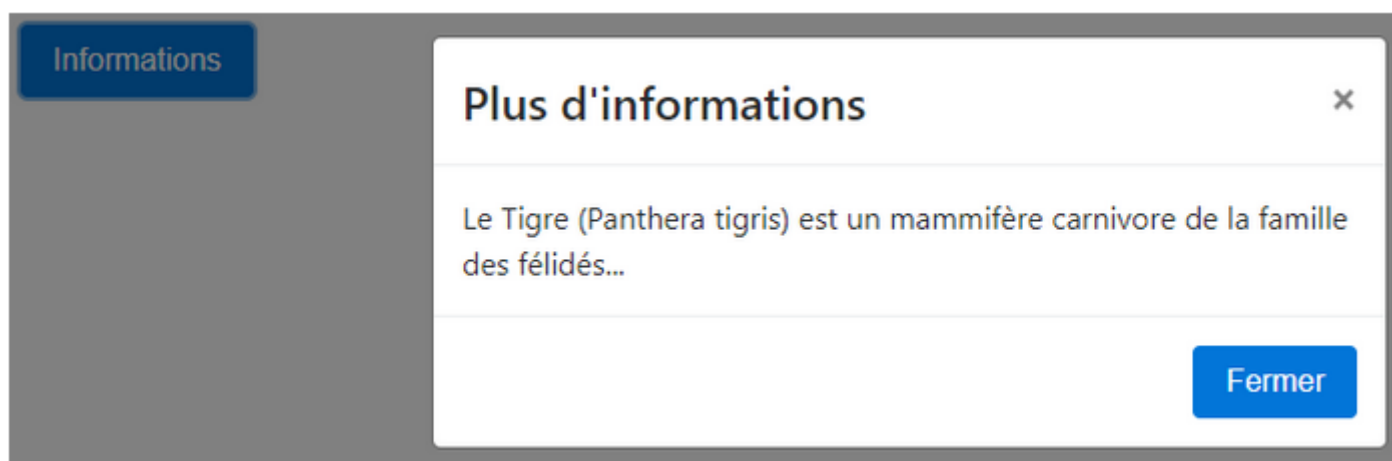
Voici l'exemple ci-dessus complété avec des commandes de fermeture :

```
<button type="button" data-toggle="modal" data-target="#infos"
class="btn btn-primary">Informations</button>
<div class="modal" id="infos">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">Plus d'informations</h4>
        <button type="button" class="close" data-dismiss="modal">
          <span>&times;</span>
        </button>
      </div>
      <div class="modal-body">
        Le Tigre (Panthera tigris) est un mammifère carnivore de
```

la famille des félidés...

```
</div>
<div class="modal-footer">
  <button type="button" class="btn btn-primary" data-
dismiss="modal">Fermer</button>
</div>
</div>
</div>
</div>
```

[Tester en ligne](#)



La classe **close** a pour but de styliser la petite croix et doit être prévue.

Dimension de la fenêtre

Il peut arriver de désirer une fenêtre plus grande ou plus étroite. On dispose de deux classes pour le réaliser :

- **modal-lg** : pour une fenêtre large
- **modal-sm** : pour une fenêtre étroite

Ces classes doivent aller avec la classe **modal-dialog**. Voici un exemple :

```
<button class="btn btn-primary" data-toggle="modal" data-
target="#f1">Grande fenêtre</button>
<div class="modal" id="f1">
  <div class="modal-dialog modal-lg">
    <div class="modal-content">
```

```

    <div class="modal-body">
      Je suis une grande fenêtre !
      <button type="button" class="close" data-dismiss="modal">
        <span>&times;</span>
      </button>
    </div>
  </div>
</div>
<button class="btn btn-primary" data-toggle="modal" data-
target="#f2">Petite fenêtre</button>
<div class="modal" id="f2">
  <div class="modal-dialog modal-sm">
    <div class="modal-content">
      <div class="modal-body">
        Je suis une petite fenêtre !
        <button type="button" class="close" data-dismiss="modal">
          <span>&times;</span>
        </button>
      </div>
    </div>
  </div>
</div>
</div>

```

Grande fenêtre avec la classe **modal-lg** :



Petite fenêtre avec la classe **modal-sm** :



Animation fluide et fond d'écran

Animation fluide

Dans les exemples vus ci-dessus la fenêtre modale apparaît d'un coup à l'écran. Il est possible de la faire apparaître avec un mouvement fluide. Pour cela il suffit d'ajouter la classe **fade** à la classe **modal**.

Prenons un exemple précédent en ajoutant cette classe :

```
<button type="button" data-toggle="modal" data-target="#infos"
class="btn btn-primary">Informations</button>
<div class="modal fade" id="infos">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal">
          <span>&times;</span>
        </button>
        <h4 class="modal-title">Plus d'informations</h4>
      </div>
      <div class="modal-body">
        Le Tigre (Panthera tigris) est un mammifère carnivore de
la famille des félidés...
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-primary" data-
dismiss="modal">Fermer</button>
      </div>
    </div>
  </div>
</div>
```

[Tester en ligne](#)



Fond d'écran

Par défaut le fond de l'écran se grise lorsque la fenêtre modale apparaît et un clic dans cette zone grisée a pour effet de fermer la fenêtre. Si vous désirez conserver une apparence normale il faut utiliser l'attribut **data-backdrop= »false »**. Par exemple pour le cas vu ci-dessus il suffit de modifier cette partie du code du bouton déclencheur :

```
<button type="button" data-toggle="modal" data-backdrop="false" data-target="#infos" class="btn btn-primary">Informations</button>
```

[Tester en ligne](#)

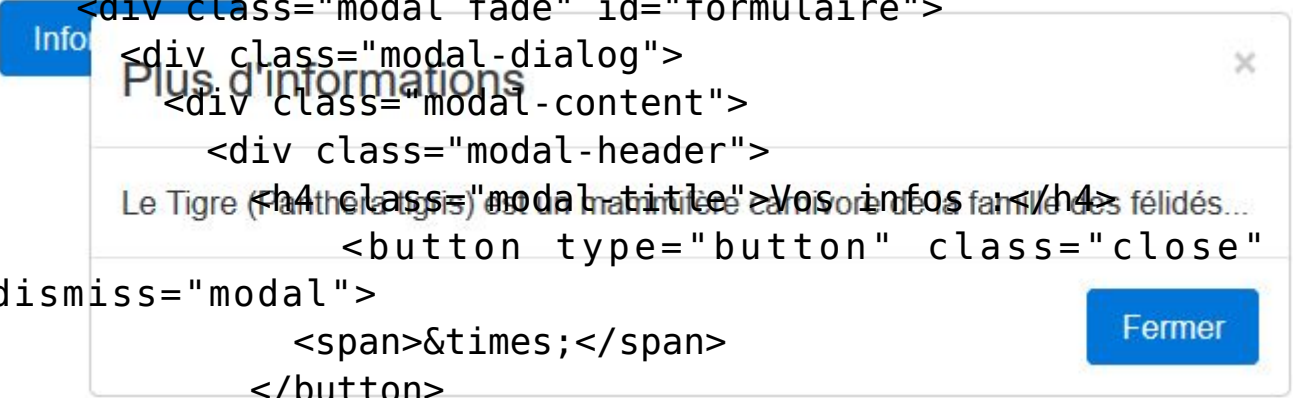
Un formulaire dans la fenêtre modale

Il est tout à fait possible d'avoir des éléments interactifs au niveau de la fenêtre modale comme par exemple un formulaire. Voici un exemple de mise en œuvre :

```
<div class="container">

  <div id="html">
    <button data-toggle="modal" data-target="#formulaire"
class="btn btn-primary">Informations</button>
  </div>

  <div class="modal fade" id="formulaire">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <h4 class="modal-title">Plus d'informations</h4>
          <button type="button" class="close" data-
dismiss="modal">
            <span>&times;</span>
          </button>
        </div>
        <div class="modal-body row">
          <form class="col" action="test.php">
            <div class="form-group">
              <label for="nom" class="form-control-
label">Nom</label>
              <input type="text" class="form-control" name
="nom" id="nom" placeholder="Votre nom">
            </div>
            <div class="form-group">
```

A screenshot of a modal dialog box. The dialog has a title bar with 'Info' on the left and a close button (X) on the right. The title is 'Plus d'informations'. The content area contains a form with a header 'Le Tigre (Panthera tigris) est un mammifère carnivore de la famille des félidés...' and a form field for 'Nom' with a placeholder 'Votre nom'. A blue button labeled 'Fermer' is at the bottom right.

```
        <label for="email" class="form-control-
label">Email</label>
        <input type="email" class="form-control"
name="email" id="email" placeholder="Votre Email">
    </div>
    <button type="submit" class="btn btn-primary pull-
right">Envoyer</button>
    </form>
</div>
</div>
</div>
</div>
```

```
</div>
```

```
<script
src="https://code.jquery.com/jquery-3.2.1.min.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.0/umd/p
opper.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script>
    $(function(){
        $('form').submit(function(e) {
            e.preventDefault()
            var $form = $(this)
            $.post($form.attr('action'), $form.serialize())
            .done(function(data) {
                $('#html').html(data)
                $('#formulaire').modal('hide')
            })
            .fail(function() {
                alert('ça ne marche pas...')
            })
        })
        $('.modal').on('shown.bs.modal', function(){
            $('input:first').focus()
        })
    })
</script>
```



Le traitement du formulaire va se faire en Ajax, de cette façon il est facile de gérer la validation (ce que je ne fais pas dans cet exemple pour ne pas alourdir le code). Il nous faut un fichier PHP (test.php) pour gérer le traitement du formulaire :

```
<?php
echo("Le nom est " . $_POST['nom'] . " et l'Email est " .
$_POST['email'] . ".");
```

Je me contente de mettre en forme les deux informations transmises et de renvoyer le tout. Ensuite j'affiche la phrase sur la page après avoir effacé la fenêtre modale (on va voir un peu plus loin dans ce chapitre comment on commande la fenêtre modale avec jQuery) :

[Tester en ligne](#)

Le nom est Dupont et l'Email est dupont@chezmoi.fr.

En fait vous pouvez mettre tout ce que vous voulez dans une fenêtre modale !

Accessibilité

Comme la fenêtre modale est une boîte de dialogue séparée du reste de la page et qui apparaît au-dessus il faut

prévoir **role= »dialog »**. Mais ce rôle n'est pas suffisant. Il faut aussi prévoir une étiquette de référence. Autrement dit répondre à la question « A quoi se rapporte cette boîte de dialogue ? ». L'idéal est d'utiliser un attribut **aria-labelledby** pour l'élément qui a le rôle **dialog**. On peut alors se référer par exemple à un titre dans la fenêtre.

Le contenu de la fenêtre lui-même se présente comme un document à consulter il faut prévoir **role= »document »**. Mais tout dépend évidemment ce que vous placez dans la fenêtre modale. Si vous prévoyez d'y mettre un formulaire alors le rôle change parce qu'il y a une interactivité.

Voici un exemple simple de réalisation :

```
<button type="button" data-toggle="modal" data-target="#infos"
class="btn btn-secondary">Commande</button>
<div class="modal" id="infos" role="dialog" aria-
labelledby="label" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title" id="label">Plus
d'informations</h4>
      </div>
      <div class="modal-body">
        Le Tigre (Panthera tigris) est un mammifère carnivore de
la famille des félidés...
      </div>
    </div>
  </div>
</div>
```

Activation avec Javascript

Ouverture

On a vu qu'on peut pratiquement tout faire avec des propriétés. Ainsi la propriété **data-toggle= »modal »** placée dans l'élément déclencheur active automatiquement le plugin. Mais vous

avez aussi la possibilité d'utiliser Javascript pour utiliser ce plugin. Reprenons un exemple vu ci-dessus en n'utilisant plus cette propriété :

```
<button type="button" class="btn btn-
primary">Informations</button>
<div class="modal" id="infos">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">Plus d'informations</h4>
        <button type="button" class="close" data-dismiss="modal">
          <span>&times;</span>
        </button>
      </div>
      <div class="modal-body">
        Le Tigre (Panthera tigris) est un mammifère carnivore de
la famille des félidés...
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-primary" data-
dismiss="modal">Fermer</button>
      </div>
    </div>
  </div>
</div>
</div>
```

Maintenant lorsqu'on clique sur le bouton, il ne se passe plus rien. Pour activer le plugin, il faut utiliser jQuery :

```
$('.btn').click(function() {
  $('.modal').modal('show')
})
```

Fermeture

De la même manière on a vu que la propriété **data-dismiss= »modal »** permet d'automatiser la fermeture. On peut utiliser jQuery pour accomplir cette action. Voilà le même exemple avec la fermeture effectuée par jQuery pour les deux boutons en plus de l'ouverture :

```
<div class="container">
```

```

    <button id="open" type="button" class="btn btn-
primary">Informations</button>
    <div class="modal" id="infos">
      <div class="modal-dialog">
        <div class="modal-content">
          <div class="modal-header">
            <h4 class="modal-title">Plus d'informations</h4>
            <button type="button" class="close closemodal">
              <span>&times;</span>
            </button>
          </div>
          <div class="modal-body">
            Le Tigre (Panthera tigris) est un mammifère carnivore de
la famille des félidés...
          </div>
          <div class="modal-footer">
            <button type="button" class="btn btn-primary
closemodal">Fermer</button>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```

<script
src="https://code.jquery.com/jquery-3.2.1.min.js"></script>
<script src="js/bootstrap.js"></script>
<script>
  $(function(){
    $('#open').click(function() {
      $('.modal').modal('show')
    })
    $('.closemodal').click(function() {
      $('.modal').modal('hide')
    })
  })
</script>

```

Les événements

Vous disposez également de 4 événements pour ce plugin :

- **show.bs.modal** : se déclenche dès l'appel à la méthode `show`. Si c'est suite à un événement clic l'élément cliqué peut être connu avec la propriété **relatedTarget**
- **shown.bs.modal** : se déclenche lorsque la fenêtre devient visible. Si c'est suite à un événement clic l'élément cliqué peut être connu avec la propriété **relatedTarget**
- **hide.bs.modal** : se déclenche dès l'appel à la méthode `hide`.
- **hidden.bs.modal** : se déclenche lorsque la fenêtre est masquée

Vous avez par exemple un formulaire sur la fenêtre modale comme on l'a vu ci-dessus et vous désirez que le premier contrôle soit actif à l'affichage de la page, vous pouvez le réaliser facilement ainsi :

```
$('.modal').on('shown.bs.modal', function(){
  $('input:first').focus()
})
```

Une autre utilisation pratique de ces événements consiste à se servir de la propriété **relatedTarget**. Supposez que vous ayez 3 boutons pour ouvrir une fenêtre modale et que pour chaque bouton le contenu de la fenêtre doit être différent, par exemple on charge une page html distincte. On pourrait évidemment utiliser 3 fenêtres totalement séparées mais on peut faire quelque chose de plus élégant :

```
<div class="container">

  <div class="btn-group">
    <button id="page1" type="button" data-toggle="modal" data-target=".modal" data-remote="1" class="btn btn-primary">Page 1</button>
    <button id="page2" type="button" data-toggle="modal" data-target=".modal" data-remote="2" class="btn btn-primary">Page 2</button>
    <button id="page3" type="button" data-toggle="modal" data-target=".modal" data-remote="3" class="btn btn-primary">Page 3</button>
  </div>
  <div class="modal fade" id="infos">
    <div class="modal-dialog">
```

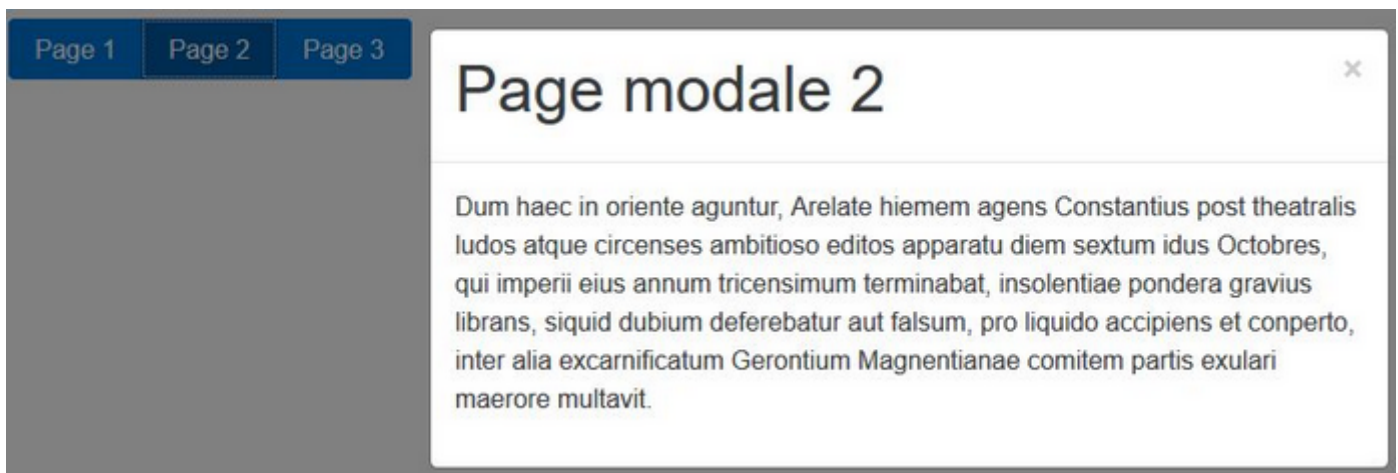
```

        <div class="modal-content"></div>
    </div>
</div>

<script
src="https://code.jquery.com/jquery-3.2.1.min.js"></script>
<script src="js/bootstrap.js"></script>
<script>
    $(function(){
        $('.modal').on('show.bs.modal', function (e) {
            var button = $(e.relatedTarget)
            var index = button.data('remote')
            $(this).find('.modal-content').load('remote' + index +
'.html')
        })
    })
</script>

```

[Tester en ligne](#)



J'ai créé un attribut **data-remote** avec les valeurs 1, 2 et 3. Lorsque l'événement **show.bs.modal** est déclenché, autrement dit dès qu'on clique sur un bouton, on peut récupérer la valeur de la propriété **data-remote** du bouton cliqué et l'utiliser pour composer le nom de la page HTML à charger.

En résumé

- Les fenêtres modales permettent de faire apparaître des informations.
- Elles peuvent comporter une entête avec éventuellement un titre, un contenu et un bas de page.
- On peut adapter la largeur et la visibilité de la page en fond.
- On peut mettre n'importe quel contenu dans une fenêtre modale, par exemple un formulaire.
- jQuery permet de créer une meilleure interactivité par exemple en mettant le focus sur un contrôle de formulaire ou en chargeant une page HTML dans la fenêtre selon le bouton cliqué.