

# LIFIHM - Interactions Humain Machine

## Programmation pour l'IHM - interfaces

Fabien Duchateau

*fabien.duchateau [at] univ-lyon1.fr*

Université Claude Bernard Lyon 1

2021 - 2022



<http://liris.cnrs.fr/stephanie.jean-daubias/enseignement/IHM/>

# Positionnement dans LIFIHM



Ces transparents utilisent le genre féminin (e.g., chercheuse, développeuses) plutôt que l'écriture inclusive (moins accessible, moins concise, et pas totalement inclusive)

# Motivation

Rappels des cours précédents :

- ▶ Méthodes de conception pour les IHM (croquis, maquette, prototype)
- ▶ Comment programmer les interfaces à partir des maquettes ?
- ▶ Comment programmer les interactions, notamment les événements déclenchés par des actions sur les composants ?

Dans cette UE, nous nous focalisons sur la programmation d'interfaces web et d'applications mobiles

# Rappels HTML

## HTML pour HyperText Markup Language :

- ▶ Langage de balisage :
  - ▶ structure (e.g., <header>, <nav>)
  - ▶ contenu (e.g., <p>, <a>, <img>, <table>)
- ▶ Standard W3C depuis 1996, version 5 en 2021
- ▶ Fichiers texte avec extension **.html**, qui sont lus et interprétés par le navigateur

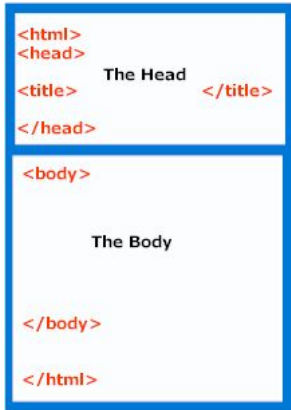
---

<http://www.w3.org/TR/html5/>

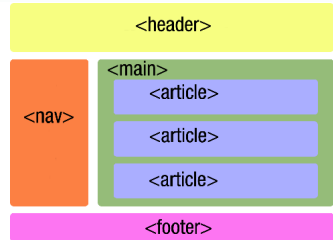
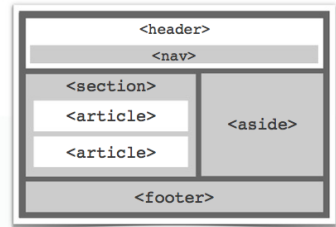
<http://www.w3schools.com/html/>

Tutoriel OpenClassRooms sur HTML5 et CSS3

## Rappels HTML (2)



*Structure générale d'un document HTML*



*Deux exemples de structuration du contenu (partie <body>)*

# Rappels HTML (3)

```
1 <!DOCTYPE html>
2 <html lang="fr">
3
4 <head>
5     <title>Page "hello world"</title>
6     <meta charset="utf-8">
7 </head>
8
9 <body>
10     <header>
11         <h1>Exemple "hello world"</h1>
12     </header>
13     <section>
14         <p>Hello world !</p>
15     </section>
16     <footer>
17         <a href="fr.wikipedia.org/wiki/Hello_world"
18             >Source</a>
19     </footer>
20 </body>
21 </html>
```

## Exemple "hello world"

Hello world !

[Source](#)

*Un exemple de fichier HTML et son résultat dans un navigateur*

# Rappels CSS

CSS pour Cascading Style Sheets :

- ▶ Langage de présentation/de rendu des données
- ▶ Standard W3C depuis 1996, version 3 en 2021
- ▶ Fichiers texte avec extension **.css**, qui sont lus et interprétés par le navigateur
- ▶ Notion de grille (*grid layout*, en recommandation candidate)

---

<http://www.w3.org/Style/CSS/>  
<http://www.w3schools.com/css/>  
<http://css.mammothland.net/>  
<http://fr.html.net/tutorials/css/>  
<http://www.w3.org/TR/css-grid-1/>

## Rappels CSS (2)

Trois manières (fréquentes) d'insérer des styles CSS :

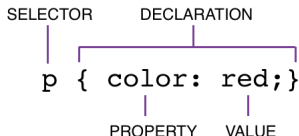
- ▶ Inline (via l'attribut `style` d'un élément)
- ▶ Interne (via une balise `<style>`)
- ▶ Externe (via une balise `<link>`)

Un style CSS se définit par :

- ▶ Un ou plusieurs sélecteurs suivis de déclarations
- ▶ Une déclaration est une paire *propriété : valeur* ;



## Rappels CSS (3)



*Structure générale d'une déclaration en CSS*

*Déclaration CSS avec  
sélecteur par balise*

```
div {  
  color: #000000;  
}
```

*Déclaration CSS avec  
sélecteur par classe*

```
.ma_classe {  
  color: #000000;  
}
```

*Déclaration CSS avec  
sélecteur par identifiant*

```
#un_ID {  
  color: #000000;  
}
```

## Rappels CSS (3)

Comment faire le lien entre une déclaration CSS et un élément HTML ?

- ▶ En insertion **inline**, accès direct à l'élément concerné
- ▶ En insertion **interne** ou **externe**, c'est le rôle du sélecteur, par exemple :
  - ▶ un sélecteur basé sur une classe (e.g., `.ma_classe` dans le CSS) s'appliquera aux éléments HTML qui ont un attribut `class` avec la valeur `.ma_classe`
  - ▶ un sélecteur basé sur un identifiant (e.g., `#un_id` dans le CSS) s'appliquera à l'élément HTML qui a un attribut `id` avec la valeur `#un_id`

# Rappels CSS (4)

```
1 header {  
2   background-color: #aa88cc;  
3   margin-bottom: 1em;  
4   width: 20em;  
5   border-radius: 0.5em;  
6 }  
7  
8 p {  
9   text-indent: 2em;  
10  word-spacing: 1em;  
11  text-transform: capitalize;  
12 }  
13  
14 footer {  
15   background-color: #55cc55;  
16   width: 20em;  
17   text-align: center;  
18 }
```

## Exemple "hello world"

Hello World !

[Source](#)

*Un style CSS et le résultat obtenu après application sur le fichier HTML précédent*

# Limitations de HTML et CSS

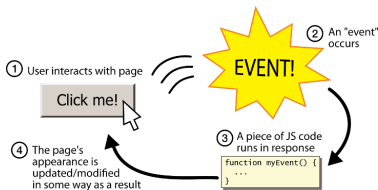
- ▶ Des langages qui évoluent rapidement, et donc nécessité de faire évoluer le code fréquemment
- ▶ Difficulté d'écrire un code qui s'adapte à tous les dispositifs existants
- ▶ Des langages pour afficher et mettre en page/forme le contenu (interfaces), mais qui ne permettent pas de programmer des événements

---

Le CSS permet de programmer quelques "événements" (e.g., au survol d'un lien), mais cela reste très limité

# Paradigme de programmation événementielle

- ▶ Le programme est défini par rapport aux événements qui peuvent survenir dans l'IHM
- ▶ Un événement étant une action utilisatrice, un message d'un autre processus, etc.
- ▶ L'interface est au service de l'utilisateur et non l'inverse



[http://fr.wikipedia.org/wiki/Paradigme\\_%28programmation%29](http://fr.wikipedia.org/wiki/Paradigme_%28programmation%29)

[http://fr.wikipedia.org/wiki/Programmation\\_%C3%A9v%C3%A8nementielle](http://fr.wikipedia.org/wiki/Programmation_%C3%A9v%C3%A8nementielle)

# Technologies pour le développement web

Interfaces :

- ▶ HTML et CSS, et des frameworks associés (e.g., Bootstrap, Foundation, Pure, KNACSS, Gumby)

Interactions côté client (programmation événementielle, RIA pour Rich Internet Applications) :

- ▶ JavaScript, Ajax, ...

Traitements côté serveur :

- ▶ PHP, Python, JavaScript, Perl, ASP, ...

---

[http://fr.wikipedia.org/wiki/Programmation\\_web](http://fr.wikipedia.org/wiki/Programmation_web)

[http://en.wikipedia.org/wiki/CSS\\_frameworks](http://en.wikipedia.org/wiki/CSS_frameworks)

[http://en.wikipedia.org/wiki/Ajax\\_framework](http://en.wikipedia.org/wiki/Ajax_framework)

# Outils pour le développement web

Éditeurs de texte (non exhaustif, multi-OS et open-source) :

- ▶ Atom (<http://atom.io/>)
- ▶ Bluefish (<http://bluefish.openoffice.nl>)
- ▶ Brackets (<http://brackets.io/>)
- ▶ Emacs (<http://www.gnu.org/software/emacs/>)
- ▶ jedit (<http://www.jedit.org/>)

Un navigateur récent (e.g., [Firefox](#), [Chrome](#)), et en option :

- ▶ Extension Firebug (<http://getfirebug.com/>)
- ▶ Un validateur HTML, un sélecteur de couleurs, des outils de mesure, des nettoyeurs de code, etc.

---

[http://fr.wikipedia.org/wiki/%C3%89diteur\\_HTML](http://fr.wikipedia.org/wiki/%C3%89diteur_HTML)

[http://fr.wikipedia.org/wiki/%C3%89diteur\\_de\\_texte](http://fr.wikipedia.org/wiki/%C3%89diteur_de_texte)

# Objectifs du cours

- ▶ Programmer des interfaces :
  - ▶ Découvrir le framework Bootstrap et son système de grille "responsive-design"
  - ▶ Savoir utiliser les styles pré-définis pour les balises HTML existantes
  - ▶ Savoir utiliser les composants supplémentaires de Bootstrap
- ▶ Programmer des événements :
  - ▶ Découvrir le Document Object Model (DOM), qui définit les propriétés et méthodes d'un document HTML
  - ▶ Connaître les types d'événements disponibles en JavaScript et programmer les actions associées



# Plan

Présentation de Bootstrap

Système de grille

Styles pré-définis

Composants supplémentaires

# Généralités

Bootstrap, une collection d'outils (framework) pour la création de sites/applications web :

- ▶ Développé initialement par Twitter (2011)
- ▶ Projet open-source, version 5 en 2021
- ▶ Utilisé côté client (front-end)
- ▶ Interfaces adaptatives (responsive-design) et homogènes



---

<http://getbootstrap.com/>

[http://fr.wikipedia.org/wiki/Bootstrap\\_%28framework%29](http://fr.wikipedia.org/wiki/Bootstrap_%28framework%29)

<http://openclassrooms.com/courses/prenez-en-main-bootstrap/>

# Installation de Bootstrap

C'est dans la partie head du fichier HTML que l'on référence :

- Soit une version locale de Bootstrap (développement)
- Soit une version de Bootstrap en ligne, via un CDN

```
13 <link href="style-exemples.css" rel="
    stylesheet">
14 <meta name="viewport" content="
    width=device-width,
    initial-scale=1, shrink-to-fit=no
">
15 <link href="lib/
    bootstrap-5.0.0-beta1-dist/css/
    bootstrap.min.css" rel="
    stylesheet">
```

```
18  Bibliothèques chargées à distance (CDN)
    pour le cours
19 <meta name="viewport" content="
    width=device-width,
    initial-scale=1">
20 <link href="https://cdn.jsdelivr.net/
    npm/bootstrap@5.0.0-beta1/dist/
    css/bootstrap.min.css" rel="
    stylesheet">
```

*Utilisation de Bootstrap (fichiers CSS et JavaScript) en local et via un CDN*

---

Le script *bootstrap.bundle.js* charge *bootstrap.js* et les autres dépendances

# Plan

Présentation de Bootstrap

**Système de grille**

Styles pré-définis

Composants supplémentaires

# Une grille adaptative

Une grille facilite l'adaptation du contenu aux différents dispositifs :

- extra small (mobiles en portrait,  $<576\text{px}$ )
- sm** small (mobiles en paysage,  $\geq 576\text{px}$ )
- md** medium (tablettes,  $\geq 768\text{px}$ )
- lg** large (ordinateurs portables,  $\geq 992\text{px}$ )
- xl** extra large (grand écrans,  $\geq 1200\text{px}$ )
- xxl** extra large (grand écrans,  $\geq 1400\text{px}$ )

Bootstrap est optimisé pour les mobiles (pas de préfixe pour xs)



---

[http://en.wikipedia.org/wiki/Responsive\\_web\\_design](http://en.wikipedia.org/wiki/Responsive_web_design)

## Système de grille (2)

Le **container** peut s'assimiler à une grille, dans laquelle on déclare des **rangées** (row). Les éléments de contenu occupent un certain nombre de **colonnes** sur une rangée donnée

Comportement de la grille :

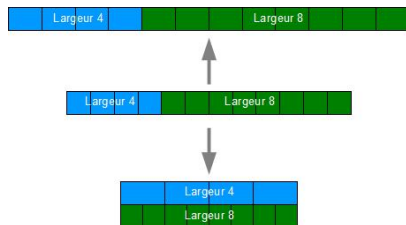
- ▶ Contenus d'abord réduits (en largeur) puis empilés (les uns sous les autres) au fur et à mesure que le dispositif rétrécit
- ▶ Possibilité de cacher des contenus quand le dispositif rétrécit
- ▶ Comportement inverse quand le dispositif s'élargit

Ce sont des **classes CSS** qui transforment des balises (e.g., *div*, *span*, *article*) en container, row et colonnes

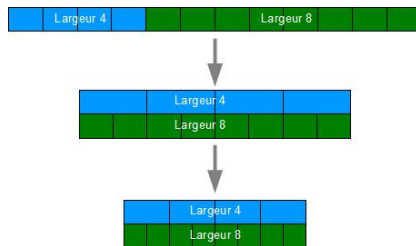
---

<http://getbootstrap.com/docs/5.0/layout/grid/>

# Exemples du système de grille



*La grille, définie pour tablettes (md, au milieu), élargit ses contenus sur des dispositifs plus larges (lg, en haut) et empile ses contenus pour des dispositifs plus petits (sm, en bas)*



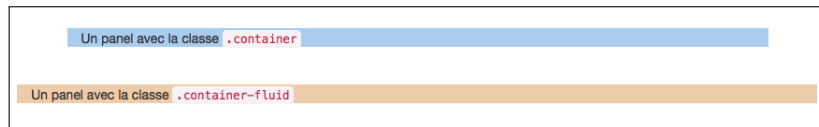
*La grille, définie pour larges dispositifs (lg, en haut), empile ses contenus sur des dispositifs de plus en plus petits (md au milieu puis sm en bas)*

<http://openclassrooms.com/courses/prenez-en-main-bootstrap/une-grille>

# Containers

Deux types de containers :

- ▶ Une classe `container` qui contient et centre la grille sur une largeur fixe (définie selon la largeur du dispositif)
- ▶ Une classe `container-fluid` qui permet à la grille d'occuper toute la largeur.



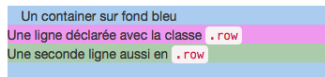
*Exemples avec les deux classes servant de container*



# Rangées

Les rangées servent à regrouper horizontalement des colonnes :

- ▶ Une seule classe, `.row`
- ▶ Les colonnes d'une rangée sont alignées correctement à partir de leur bord haut, et les contenus volumineux s'écoulent donc verticalement
- ▶ Une rangée occupe la hauteur du contenu de sa plus grande colonne



*Exemples avec la classe `.row`*

# Colonnes

Le contenu se déclare à l'intérieur de colonnes :

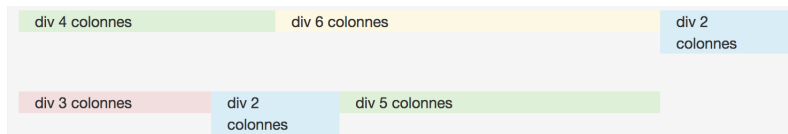
- ▶ Les colonnes sont les seuls éléments "enfant" des rangées
- ▶ Par défaut dans Bootstrap, on dispose de 12 colonnes par rangée
- ▶ La taille des colonnes est automatiquement calculée (selon un pourcentage), ce qui permet l'adaptation à la taille de l'écran
- ▶ Le contenu a des marges droite et gauche de 15px

## Colonnes (2)

- ▶ La classe indique le nombre de colonnes occupées par le contenu (jusqu'à 12 colonnes) :
  - ▶ `.col-1` signifie que le contenu occupe 1 colonne en xs
  - ▶ `.col-sm-4` signifie que le contenu occupe 4 colonnes en sm
  - ▶ `.col-md-4` signifie que le contenu occupe 4 colonnes en md
  - ▶ ...
- ▶ Les dispositifs sont définis selon des largeurs minimales, donc une classe de colonne s'applique à ce dispositif **et aux dispositifs plus grands**
  - ▶ un élément défini par la classe `.col-sm-8` aura un contenu sur 8 colonnes pour les dispositifs sm, md, lg et xl

# Exemple de grille

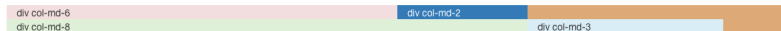
```
2 <div class="row">
3   <div class="col-4 bg-success">div 4 colonnes</div>
4   <div class="col-6 bg-warning">div 6 colonnes</div>
5   <div class="col-2 bg-info">div 2 colonnes</div>
6 </div>
7 <div class="row">
8   <div class="col-3 bg-danger">div 3 colonnes</div>
9   <div class="col-2 bg-info">div 2 colonnes</div>
10  <div class="col-5 bg-success">div 5 colonnes</div>
11 </div>
```



*Deux rangées dont les éléments (colorés avec les classes `.bg-*`) occupent un certain nombre de colonnes en xs*

# Enroulement de colonnes

Quand un élément (réparti sur plusieurs colonnes) n'a plus assez de place disponible, il est placé sur une ligne en dessous (mais toujours dans la même rangée)



*Une rangée (fond marron) avec quatre éléments. Le troisième élément qui occupe 8 colonnes n'a pas assez de place pour se mettre sur la première ligne ( $12 - 6 - 2 < 8$ ) et se retrouve donc sur une nouvelle ligne*

# Réorganisation de colonnes

Par défaut, les colonnes sont affichées dans l'ordre du flux, mais réorganisation possible avec les classes :

- ▶ `.offset-*-x` pour déplacer un élément à droite de x colonnes ( $\approx$  marge à gauche)
- ▶ `.order-*-x` pour réordonner visuellement en indiquant la position de la colonne (avec x variant de 1 à 12)



col-md-5

col-md-5 col-md-offset-2

*Sur cette rangée (verte), le second élément (bleu) possède un offset de 2, donc on saute deux colonnes avant de l'insérer*

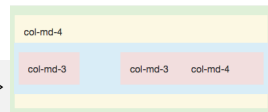
---

<http://getbootstrap.com/docs/5.0/layout/columns/#reordering>

# Imbrication d'éléments

Dans un groupe de colonnes, déclarer un élément de classe `.row`, qui peut aussi contenir des groupes de colonnes

```
2 <div class="row bg-success padding-20">
3   <div class="col-md-4 bg-warning padding-10 margin-bottom-05">
4     col-md-4
5     <div class="row bg-info">
6       <div class="col-md-3 bg-danger padding-10">col-md-3</div>
7       <div class="col-md-3 bg-danger col-lg-offset-2 padding-10">col-md-3</div>
8       <div class="col-md-4 bg-danger padding-10">col-md-4</div>
9     </div>
10  </div>
```

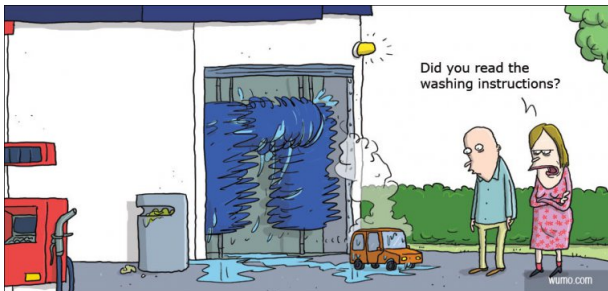


*Une première rangée qui inclut un groupe de 4 colonnes. Ce groupe de 4 colonnes contient une nouvelle rangée, dans laquelle on ajoute plusieurs groupes de colonnes (code tronqué)*

# En résumé

Bootstrap, un framework CSS populaire et bien documenté :

- ▶ Système de grille (12 colonnes et adaptatif)
- ▶ `.container > .row (rangée) > .col-*-x` (groupes de colonnes)
- ▶ Enroulement, sauts de colonnes, imbrication, décalages





# Plan

Présentation de Bootstrap

Système de grille

**Styles pré-définis**

Composants supplémentaires

# Des styles CSS pré-définis

Bootstrap définit des styles pour certaines balises :

- ▶ Mise en forme esthétique, sans code CSS à écrire

Conseils :

- ▶ Éviter d'utiliser des marges en haut (`margin-top`), dont le comportement est parfois imprévisible
- ▶ Privilégier l'unité `rem` (*root em*)

Par convention dans ce cours :

- ▶ `<x>` indique une balise (éventuellement à fermer)
- ▶ `.x` indique une classe
- ▶ `#x` indique un identifiant
- ▶ `*` indique un dispositif (parmi `xs`, `sm`, `md`, `lg`, `xl`, `xxl`)

# Valeurs par défaut pour la page

- ▶ Taille de police = 16px (soit 1 rem)
- ▶ Utilisation de `box-sizing: border-box`; pour inclure paddings et bordures dans la largeur et hauteur des éléments
  - ▶ évite les éléments trop larges à cause d'un padding ou d'une marge excessive
- ▶ Fond blanc
- ▶ Ajout/suppression de marges pour les balises de contenu (e.g., `<h1>` et `<p>` ont une marge en bas)
- ▶ ...

---

<http://getbootstrap.com/docs/5.0/content/reboot/>

# Classes utilitaires - texte

- ▶ `<em>` (*italique*), `<strong>` (**gras**), `<mark>` (fond jaune), `<small>` (taille réduite), `<u>` (souligné), ...
- ▶ `.h1`, ..., `.h6` pour un texte avec l'apparence d'un titre (mais en inline), `.display-1`, ..., `.display-6`
- ▶ `<abbr title="explication">`, `<code>`,
- ▶ `.lead` pour un texte en évidence, `.text-lowercase`, `.text-uppercase`, `.text-capitalize`
- ▶ `.fs-1`, ..., `.fs-6` pour la taille de police, `.fw-bold`, etc.

Des exemples de texte avec balise `em`, de texte avec balise `strong`, de texte avec balise `em`, de texte avec balise `small`, de texte avec balise `em`, de texte avec balise `del`.

Un autre exemple avec un texte qui a le style d'un titre `h3`, une abréviation `CSS`, et enfin des textes transformées par des classes prédéfinies : le premier avec `text-lowercase`, le SECOND AVEC `TEXT-UPPERCASE` et le Troisième Avec `Text-capitalize`.

---

<http://getbootstrap.com/docs/5.0/content/typography/>  
<http://getbootstrap.com/docs/5.0/utilities/text/>

# Classes utilitaires - couleurs de texte

- ▶ `.text-primary` pour du texte important
- ▶ `.text-secondary` pour du texte secondaire
- ▶ `.text-success` pour une confirmation (e.g., succès)
- ▶ `.text-info` pour une information
- ▶ `.text-warning` pour un avertissement
- ▶ `.text-danger` pour un danger (e.g., erreur)
- ▶ `.text-muted` pour du texte grisé

```
4 <span class="text-primary">Un premier avec .text-primary.</span>  
5 <span class="text-success">En deux, un .text-success.</span>  
6 <span class="text-info">On continue avec .text-info.</span>
```

Un premier avec `.text-primary`. En deux, un `.text-success`. On continue avec `.text-info`. Ici un avertissement avec `.text-warning`. Un danger avec `.text-danger`. Enfin, un `.text-muted`.

# Classes utilitaires - couleurs de fond

- ▶ `.bg-primary` pour un fond bleu foncé
- ▶ `.bg-secondary` pour un fond gris
- ▶ `.bg-success` pour un fond vert
- ▶ `.bg-info` pour un fond bleu ciel
- ▶ `.bg-warning` pour un fond jaune
- ▶ `.bg-danger` pour un fond rouge
- ▶ `.bg-light` pour un fond gris clair

```
13 <span class="bg-primary">Un  
    fond avec .  
    bg-primary.</span>  
14 <span class="bg-success">Un  
    autre utile pour les  
    messages de  
    confirmation, avec  
    bg-success.</span>  
15 <span class="bg-info">Un  
    troisième informatif  
    avec .bg-info.</span>  
16 <span class="bg-warning">Un  
    fond .bg-warning.</s  
    pan>
```

Un fond avec `.bg-primary`, Un autre utile pour les messages de confirmation, avec `bg-success`, Un troisième informatif avec `.bg-info`, Un fond `.bg-warning`, Un fond pratique pour signaler des erreurs (`.bg-danger`), Un fond discret `.bg-light`, Enfin un fond `.bg-secondary`

# Classes utilitaires - alignement

- ▶ `.text-start`, `.text-end`, `.text-center`
  - ▶ `.text-*-start`, `.text-*-end` et `.text-*-center`
  - ▶ par exemple, `.text-start` pour aligner à gauche sur tous les dispositifs ou `.text-md-center` pour centrer sur les dispositifs moyens et supérieurs



- ▶ `.text-wrap` ou `.text-nowrap`

# Classes utilitaires - divers

- ▶ `.visible`, `.invisible` pour rendre (in)visible un élément
- ▶ `.close` pour une petite croix fermante (e.g., sur les panels), mais besoin d'associer un événement
- ▶ `.visually-hidden` pour afficher des éléments uniquement pour les outils de synthèse vocale (accessibilité)
- ▶ Des classes pour faciliter les marges et paddings (e.g., `.mt-2` pour margin top réduite de moitié, `.p-md-auto` pour tous les paddings réglés automatiquement sur dispositifs *md*)

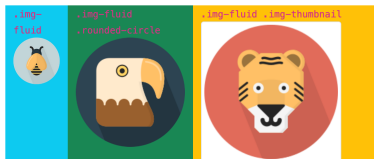
---

<http://getbootstrap.com/docs/5.0/utilities/>  
<http://getbootstrap.com/docs/5.0/utilities/spacing/>



# Images (<img>)

- ▶ `.img-fluid` garantit que l'image s'adapte et ne déborde pas de son parent
- ▶ `.rounded` (coins arrondis), `.img-thumbnail` (aperçu)
- ▶ Alignement d'une image avec `.text-center`, `.text-end`, ... dans le conteneur parent



*Une image sur 1 colonne, la seconde arrondie sur 2 colonnes  
et la dernière en thumbnail sur 3 colonnes*

---

<http://getbootstrap.com/docs/5.0/content/images/>  
<http://getbootstrap.com/docs/5.0/utilities/borders/>

# Liens et boutons

Balises `<a>`, `<button>` et `<input type="submit">` :

- ▶ `.btn` et `.btn-link` pour une apparence de bouton ou de lien
- ▶ Coloration avec couleurs contextuelles :
  - ▶ `.btn-primary` pour une action principale
  - ▶ `.btn-secondary` pour une action secondaire
  - ▶ `.btn-success` pour une action positive / réussite
  - ▶ `.btn-info` pour une action informative
  - ▶ `.btn-warning` pour une action avec précaution
  - ▶ `.btn-danger` pour une action sensible
- ▶ Coloration alternative avec `.btn-outline-primary`, etc.



---

<http://getbootstrap.com/docs/5.0/components/buttons/>

## Liens et boutons (2)

- ▶ `.btn-block` pour un bouton sur toute la largeur
- ▶ `.btn-lg` et `.btn-sm` pour un bouton plus grand ou plus petit
- ▶ `.active` et `.disabled` pour un bouton actif ou désactivé
- ▶ `.btn-group` et `.btn-group-toggle` pour les boutons radio, cases à cocher, groupes de boutons, ...

bouton `.btn` `.btn-info` `.btn-block`

Lien activé (`.active`)

Bouton inactif (attribut `disabled`)

Lien inactif (`.disabled`)

Marche

Vélo

Transport en commun

---

<http://getbootstrap.com/docs/5.0/components/button-group/>

# Listes (<ul>, <ol> et <dl>)

- ▶ `.list-unstyled` pour supprimer la marge et la puce d'une liste
- ▶ `.list-inline` pour une liste sur une ligne :
  - ▶ `.list-inline-item` pour les items <li>
- ▶ description <dl> avec éléments (dt et dd) alignés via la grille

item1  
item2  
item3

item1 item2 item3

<b>item 1</b>	description 1
<b>item 2</b>	description 2a description 2b
<b>item 3, d...</b>	description 3 (avec <code>.text-truncate</code> sur l'item)

*Une liste sans style (gauche), une liste en ligne (milieu) et une description (droite)*

# Tableaux (<table>)

- ▶ `.table` pour un style de base (padding et lignes horizontales)
- ▶ `.table-responsive` pour les petits dispositifs (ascenseur horizontal)
- ▶ `.table-bordered` pour des bordures autour de chaque cellule
- ▶ `.table-sm` pour un tableau compact
- ▶ `.table-striped` pour des lignes zébrées
- ▶ `.table-hover` pour un effet au survol de la souris

---

<http://getbootstrap.com/docs/4.0/content/tables/>

# Tableaux (2)

- Classes contextuelles `.table-active`, `.table-success`, `.table-info`, `.table-warning` et `.table-danger` applicables aux lignes ou aux cellules

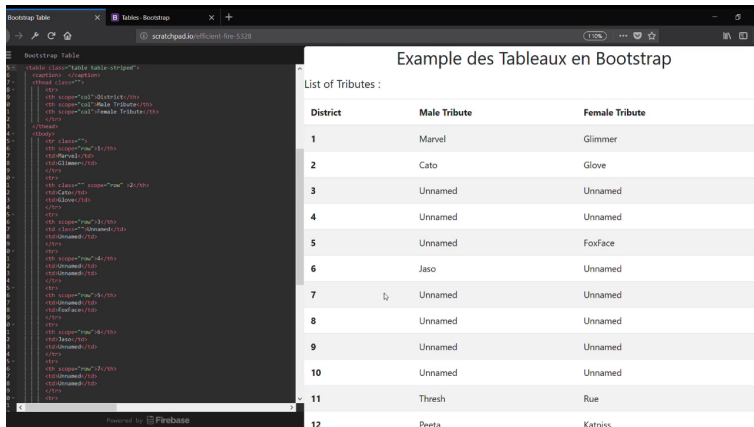
Tableau `.table` `.table-bordered` `.table-striped` placé dans une div `.table-responsive`

#	Nom	Espèce
1	Aline	Aigle
2	Ben	Bouc
3	Carole	Crocodile

Tableau `.table` `.table-condensed` `.table-hover` `.table-dark`

#	Nom	Espèce
1	Aline	Aigle
2	Ben	Bouc
3	Carole	Crocodile

# Tableaux - tutoriel



```
Bootstrap Table
<table class="table table-striped">
  <caption> </caption>
  <thead class="">
    <tr>
      <th scope="col">District</th>
      <th scope="col">Male Tribute</th>
      <th scope="col">Female Tribute</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>1</td>
      <td>Marvel</td>
      <td>Glimmer</td>
    </tr>
    <tr>
      <td>2</td>
      <td>Cato</td>
      &td>Glove</td>
    </tr>
    <tr>
      <td>3</td>
      <td>Unnamed</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>4</td>
      <td>Unnamed</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>5</td>
      <td>Unnamed</td>
      &td>FoxFace</td>
    </tr>
    <tr>
      <td>6</td>
      <td>Jaso</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>7</td>
      <td>Unnamed</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>8</td>
      <td>Unnamed</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>9</td>
      <td>Unnamed</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>10</td>
      <td>Unnamed</td>
      &td>Unnamed</td>
    </tr>
    <tr>
      <td>11</td>
      <td>Thresh</td>
      &td>Rue</td>
    </tr>
    <tr>
      <td>12</td>
      <td>Peeta</td>
      &td>Katniss</td>
    </tr>
  </tbody>
</table>
```

Exemple des Tableaux en Bootstrap

List of Tributes :

District	Male Tribute	Female Tribute
1	Marvel	Glimmer
2	Cato	Glove
3	Unnamed	Unnamed
4	Unnamed	Unnamed
5	Unnamed	FoxFace
6	Jaso	Unnamed
7	Unnamed	Unnamed
8	Unnamed	Unnamed
9	Unnamed	Unnamed
10	Unnamed	Unnamed
11	Thresh	Rue
12	Peeta	Katniss

<http://www.youtube.com/watch?v=XRSgvvpB2Fc>

# Formulaires (<form>)

- ▶ Formulaire par défaut vertical, `.form-inline` pour un formulaire horizontal
- ▶ Alignement avec la grille ou par regroupement (`.form-group`)

```
3 <form>
4   <fieldset class="form-group">
5     <label for="input1">Un premier label</label>
6     <input type="text" class="form-control" id="input1" placeholder="un exemple
7       de valeur">
8   </fieldset>
9   <fieldset class="form-group">
10    <label for="input2">Un autre label</label>
11    <input type="text" class="form-control" id="input2" placeholder="une autre
12      exemple">
13 </form>
```

Un premier label

Un autre label

<http://getbootstrap.com/docs/5.0/forms/overview/>



# Formulaires (2)

- ▶ `.form-control` pour un contrôle (`<input>`, `<select>`, `<textarea>`), `.form-control-label` pour son label
- ▶ `.is-valid`, `.has-warning` et `.is-invalid` pour indiquer un succès ou une erreur sur un composant ou un groupe

Prénom  Email

Un contrôle `.has-success`

Un contrôle `.has-warning`

Un contrôle `.has-danger`

☐ Case à cocher `.has-success`

☐ Case à cocher `.has-warning`

☐ Case à cocher `.has-danger`

Email

Identifiant

Options

☒ Option 1 (cochée par défaut)

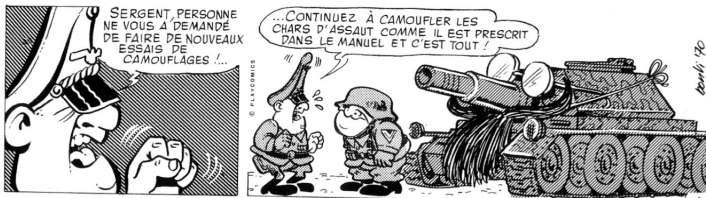
☐ Option 2 (désactivée)

Supplément

☐ Une option incontournable ! ☐ Une option moins bien

# En résumé

- ▶ Mise en forme des éléments HTML classiques par Bootstrap :
  - ▶ Listes, tableaux, images, liens/boutons, etc.
  - ▶ Classes utilitaires (e.g., couleurs, alignement)
  - ▶ Formulaires
- ▶ Possibilité de personnaliser ces styles (propre CSS, thèmes ou avec LESS/SASS)



# Plan

Présentation de Bootstrap

Système de grille

Styles pré-définis

Composants supplémentaires

# Composants Bootstrap

HTML ne fournit pas de composants "sophistiqués", mais :

- ▶ Une vingtaine de nouveaux composants dans Bootstrap
- ▶ Des centaines disponibles (e.g., formulaires d'authentification, tableaux de prix, etc.) dans les extensions

Dans ce cours, 10 composants :

- ▶ Conteneur (panels, alertes, fenêtres modales)
- ▶ Navigation (menus, barres de menus, fil d'Ariane)
- ▶ Informations secondaires (labels, infobulles, repli)

---

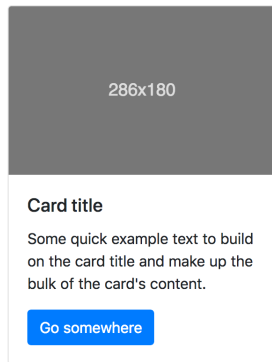
<http://wrapbootstrap.com/>

<http://startbootstrap.com/bootstrap-resources/>

# Panels (card)

Un panel est un conteneur flexible et structuré :

- ▶ `.card` pour déclarer le container
- ▶ `.card-header` et `.card-footer`
- ▶ `.card-body` pour le contenu
- ▶ `.card-img-top` et `.card-img-bottom`



---

<http://getbootstrap.com/docs/5.0/components/card/>

# Panels (card) - contenu

- ▶ .card-title et .card-subtitle
- ▶ .card-text, .card-link, .list-group, etc.
- ▶ Alignement, bordures, couleurs, onglets, etc.

```
33 <div class="row">
34 <div class="card bg-light col-3">
35   <div class="card-img-top"></div>
36   <div class="card-body">
37     <h4 class="card-title">Pull en laine de licorne</h4>
38     <p class="card-text">Idéal pour les hivers rugueux, un pull en laine de
      licorne préhistorique !</p>
39     <a href="https://www.futura-sciences.com/sciences/science-decalee/" class="
      card-link">En savoir plus sur la licorne</a>
40   </div>
41 </div>
42 <div class="card bg-light col-3 offset-1">
43   <div class="card-img-top"></div>
44   <div class="card-body">
45     <h4 class="card-title">Chaussettes en fourrure de lorax</h4>
46     <p class="card-text">Idéal pour les hivers rugueux, un pull en laine de
      licorne préhistorique !</p>
47     <a href="https://www.futura-sciences.com/sciences/science-decalee/" class="
      card-link">En savoir plus sur le lorax</a>
48   </div>
```

<http://www.futura-sciences.com/planete/zoologie-science-decalee-62252>

# Panels (card) - contenu

- ▶ `.card-title` et `.card-subtitle`
- ▶ `.card-text`, `.card-link`, `.list-group`, etc.
- ▶ Alignement, bordures, couleurs, onglets, etc.

```

33 <div class="row">
34 <div class="card bg-light col-3">
35   <div class="card-img-top"></div>
36   <div class="card-body">
37     <h4 class="card-title">Pull en laine de licorne</h4>
38     <p class="card-text">Idéal pour les hivers rugueux, un pull en laine de
      licorne préhistorique !</p>
39     <a href="https://www.futura-sciences.com/sciences/sci
      card-link">En savoir plus sur la licorne</a>
40   </div>
41 </div>
42 <div class="card bg-light col-3 offset-1">
43   <div class="card-img-top"></div>
44   <div class="card-body">
45     <h4 class="card-title">Chaussettes en fourrure de lorax</h4>
46     <p class="card-text">Idéal pour les hivers rugueux, un pull en laine
      de licorne préhistorique !</p>
47     <a href="https://www.futura-sciences.com/sciences/sci
      card-link">En savoir plus sur le lorax</a>
48   </div>

```



#### Pull en laine de licorne

Idéal pour les hivers rugueux, un pull en laine de licorne préhistorique !

[En savoir plus sur la licorne](#)



#### Chaussettes en fourrure de lorax

Idéal pour les hivers rugueux, un pull en laine de licorne préhistorique !

[En savoir plus sur le lorax](#)

<http://www.futura-sciences.com/planete/zoologie-science-decalee-62252>

# Alertes (alert)

Une alerte fournit des messages contextuels "inline" à l'utilisatrice :

- ▶ `.alert` pour les bordures et marges
- ▶ `.alert-primary`, `.alert-info`, `.alert-warning`, `.alert-success`, `.alert-danger`, etc.
- ▶ `.alert-link` pour des liens colorés selon le type d'alerte
- ▶ `.alert-heading` pour un titre, `<hr>` pour un séparateur

**Info pratique.** Au loin vous distinguez un dragon.

## Attention !

Le dragon est de taille impressionnante, et de plus en plus proche. Il ne faudrait pas avancer davantage.

---

<http://getbootstrap.com/docs/5.0/components/alerts/>



# Alertes (alert) - options

- ▶ `.alert-dismissible` pour pouvoir fermer l'alerte :
  - ▶ la croix est un bouton (dans l'alerte) avec les attributs `class="btn-close"` et `data-bs-dismiss="alert"`
  - ▶ classes `.fade` et `.show` pour la transition

```
9 <div class="alert alert-danger
    alert-dismissible fade show" role="alert"
10 >
    <button type="button" class="btn-close"
        data-bs-dismiss="alert" aria-label="
11 Fermer"></button>
    <strong>Danger ! </strong>Le dragon vous
        attaque !
12 <hr>Fuuuuuuuuez !
13 </div>
14 <div class="alert alert-success" role="alert">
15 <strong>Ouf !</strong> Vous avez <a href="
    https://www.youtube.com/watch?
        v=jvwg1b30rsw" target="_blank" class=
        "alert-link">échappé au dragon</a> à
        couvert sous la forêt.
16 </div>
```

**Danger !** Le dragon vous attaque !



Fuuuuuuuuez !

Ouf ! Vous avez échappé au dragon à couvert sous la forêt.

# Fenêtres modales (modal)

Une fenêtre modale est une fenêtre bloquante (au premier plan) :

- ▶ Généralement une balise `<div>` avec la classe `.modal`
- ▶ Les modales de Bootstrap se ferment automatiquement quand on clique en dehors de la fenêtre modale

Le composant qui ouvre la fenêtre modale (e.g., un bouton) doit :

- ▶ Avoir un attribut `data-bs-toggle="modal"` pour indiquer qu'il déclenche une fenêtre modale
- ▶ Avoir un attribut `data-bs-target="#id_modale"` qui spécifie la fenêtre modale à lancer (ici grâce à l'identifiant de la fenêtre)

---

<http://getbootstrap.com/docs/5.0/components/modal/>

# Fenêtres modales (modal) - structure

Dans une fenêtre modale (avec la classe `.modal`), plusieurs `div` :

- ▶ `.modal-dialog` spécifie la taille de fenêtre et contient :
  - ▶ `.modal-content` précise la couleur de fond et peut contenir :
    - ▶ `.modal-header` (e.g., la croix de fermeture ou le titre, spécifié avec `.modal-title`)
    - ▶ `.modal-body` (e.g., pour le texte principal de la modale)
    - ▶ `.modal-footer` (e.g., pour les boutons d'interaction)

Nombreuses options : taille, agencement, scrolling, animation, réutilisation d'une même modale avec contenu différent, etc.



# Exemple de fenêtre modale

```
3 <!-- bouton de lancement de la fenêtre modale -->
4 <button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="
  #exempleModal1">Cliquez-moi !</button>
5 <!-- déclaration de la fenêtre modale -->
6 <div id="exempleModal1" class="modal fade" tabindex="-1" role="dialog">
7   <div class="modal-dialog modal-sm" role="document">
8     <div class="modal-content">
9       <div class="modal-header">
10         <h4 class="modal-title">Le titre de la modale</h4>
11         <button type="button" class="btn btn-close" data-bs-dismiss="modal"
12           aria-label="Close"></button>
13       </div>
14       <div class="modal-body">
15         Contenu de la modale (dans <code>.modal-body</code>)
16       </div>
17       <div class="modal-footer">
18         <button type="button" class="btn btn-secondary" data-bs-dismiss="
19           modal">Annuler</button>
20         <button type="button" class="btn btn-primary">Valider</button>
21       </div>
22     </div>
  </div>
</div>
```

# Exemple de fenêtre modale

```
3 <!-- bouton de lancement de la fenêtre modale -->
4 <button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="
  #exempleModal1">Cliquez-moi !</button>
5 <!-- déclaration de la fenêtre modale -->
6 <div id="exempleModal1" class="modal fade" tabindex="-1" role="dialog">
7   <div class="modal-dialog modal-sm" role="document">
8     <div class="modal-content">
9       <div class="modal-header">
10        <h4 class="modal-title">Le titre de la modale</h4>
11        <button type="button" class="btn btn-close" data-bs-dismiss="modal"
          aria-label="Close"></button>
12      </div>
13      <div class="modal-body">
14        Contenu de la modale (dans <code>.modal-body</code>)
15      </div>
16      <div class="modal-footer">
17        <button type="button" class="btn btn-secondary" data-bs-dismiss="
          modal">Annuler</button>
18        <button type="button" class="btn btn-primary">Valider</button>
19      </div>
20    </div>
21  </div>
22 </div>
```



# Menus (nav)

La classe `.nav` sert à représenter une zone de navigation (menu) :

► Sous forme de liste (horizontale par défaut)



- `.flex-column` (verticale), `.nav-fill` (étirée)
- `.nav-item` pour les éléments de la liste
- `.nav-link` pour des liens, `.active` et `.disabled` pour un état actif ou désactivé



► Sous forme d'onglets avec `.nav-tabs`



► Sous forme de "pills" avec `.nav-pills`



<http://getbootstrap.com/docs/5.0/components/navs-tabs/>

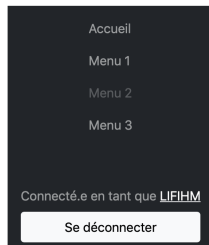
# Barre de menus (navbar)

Une barre de menus peut contenir un logo, un menu, une barre de recherche ou un formulaire, etc.

- ▶ `.navbar` pour la barre
- ▶ `.navbar-brand` pour la marque ou le logo
- ▶ `.navbar-nav` pour le menu

Options :

- ▶ Position (`.fixed-top`, `.sticky-top`, etc.)
- ▶ Couleur (`.navbar-dark`, `.navbar-light`, etc.)
- ▶ Responsive (avec `.navbar-toggler` et `collapse`)



---

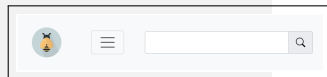
<http://getbootstrap.com/docs/5.0/components/navbar/>

# Exemple de barre de menus

```

3 <nav class="nav navbar navbar-expand-xl bg-light navbar-light">
4 <div class="container-fluid">
5   <span class="navbar-brand">
6     <a href="#"></a>
7   </span>
8   <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
9     data-bs-target="#navbarMenu" aria-controls="navbarSupportedContent"
10    aria-expanded="false" aria-label="Toggle navigation">
11     <span class="navbar-toggler-icon"></span>
12   </button>
13   <div class="collapse navbar-collapse" id="navbarMenu">
14     <ul class="navbar-nav me-auto mb-2 mb-lg-0">
15       <li class="nav-item active"><a href="#" class="nav-link">Menu 1</a></li>
16       <li class="nav-item"><a href="#" class="nav-link">Menu 2</a></li>
17       <li class="nav-item"><a href="#" class="nav-link">Menu 3</a></li>
18     </ul>
19   </div>
20   <form class="d-flex">
21     <div class="input-group">
22       <input type="search" id="query" class="form-control">
23       <button class="input-group-text" type="button"></button>
24     </div>
25   </form>
26 </div>
27 </nav>

```





# Fil d'Ariane (breadcrumb)

Un fil d'Ariane précise la localisation dans le site :

- ▶ Liste (<ol>) avec la classe `.breadcrumb`
- ▶ Éléments (<li>) avec la classe `.breadcrumb-item`, généralement des liens
- ▶ Catégorie courante avec la classe `.active`
- ▶ Seuls les séparateurs sont ajoutés (le reste est à gérer autrement)

```
2 <ol class="breadcrumb">
3   <li class="breadcrumb-item"><a href="#"
   >Accueil</a></li>
4   <li class="breadcrumb-item"><a href="#"
   >Articles magiques</a></li>
5   <li class="breadcrumb-item active">Parchemins
   </li>
6 </ol>
```

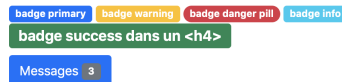
[Accueil](#) / [Articles magiques](#) / Parchemins

# Étiquettes (badge)

Les badges mettent en avant une information courte, ponctuelle :

- ▶ Classe `.badge` et une classe de couleur ( `.bg-primary`, `.bg-success`, `.bg-danger`, etc.)
- ▶ Taille de police de son élément parent
- ▶ Classe `.rounded-pill` pour des labels arrondis

```
7 <span class="badge bg-primary">badge primary</span>
8 <span class="badge bg-warning">badge warning</span>
9 <span class="badge rounded-pill bg-danger">badge
  danger pill</span>
10 <span class="badge bg-info">badge info</span>
11 <h4><span class="badge bg-success">badge success dans
  un <h4></span></h4></h4>
12 <button type="button" class="btn btn-primary">
13   Messages <span class="badge bg-secondary">3</span>
14 </button>
```



# Infobulle (tooltip)

Une infobulle permet d'afficher des informations complémentaires :

- ▶ Nécessite la bibliothèque *popper.js* (incluse dans *bundle*)
- ▶ Initialisation des tooltips (individuelle ou globale)
- ▶ Tout élément (visible et non désactivé) peut avoir un tooltip en précisant :
  - ▶ un attribut `data-bs-toggle="tooltip"`
  - ▶ un attribut `title` non vide (qui peut contenir du code HTML)

```
3 <script>
4 /* code JS pour initialiser tous les tooltips sur la page */
5 var tooltipElementList = [].slice.call(document.querySelectorAll('[data-bs-toggle="
6     tooltip"]'));
7 var tooltipList = tooltipElementList.map(function (tooltipElem) {
8     return new bootstrap.Tooltip(tooltipElem);
9 });
10 </script>
```

---

<http://getbootstrap.com/docs/5.0/components/tooltips/>

# Infobulle (tooltip) - options

- Positionnement du tooltip avec l'attribut `data-placement="left | right | top | bottom"`
- Autres options disponibles (e.g., délai et durée d'apparition, animation, container)

```
11 <button type="button" class="btn btn-info"  
12     data-bs-toggle="tooltip" data-bs-placement="top"  
13     title="Coucou, tooltip vers le haut !">  
    Un tooltip en haut  
</button>
```



```
20 
```



# Infobulle (tooltip) - tutoriel



---

<http://www.youtube.com/watch?v=S4Hv9VD4T6o>

# Repli (collapse)

Un repli permet de masquer/afficher du contenu sur demande :

- ▶ Un élément déclencheur (e.g., un bouton) possède :
  - ▶ un attribut `data-bs-toggle="collapse"`
  - ▶ un attribut `data-bs-target` valué à un sélecteur CSS
- ▶ Un élément repliable (e.g., un panel) possède :
  - ▶ la classe `.collapse`
  - ▶ en option, la classe `.show` (déplié initialement)

```
24 <a data-bs-toggle="collapse" href="#collapseDiv1">Lien déclencheur</a>
25 <div class="collapse bg-success" id="collapseDiv1">
26   Voilà le contenu replié et repliable pour le lien !
27 </div>
```

[Lien déclencheur](#)

---

<http://getbootstrap.com/docs/5.0/components/collapse/>

# Repli (collapse)

Un repli permet de masquer/afficher du contenu sur demande :

- ▶ Un élément déclencheur (e.g., un bouton) possède :
  - ▶ un attribut `data-bs-toggle="collapse"`
  - ▶ un attribut `data-bs-target` valué à un sélecteur CSS
- ▶ Un élément repliable (e.g., un panel) possède :
  - ▶ la classe `.collapse`
  - ▶ en option, la classe `.show` (déplié initialement)

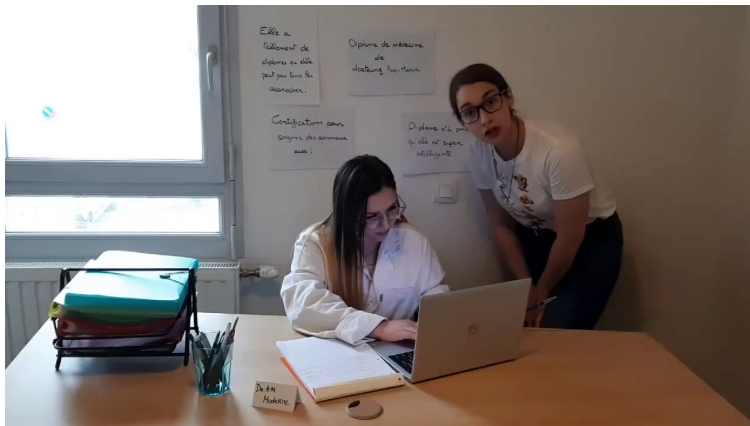
```
24 <a data-bs-toggle="collapse" href="#collapseDiv1">Lien déclencheur</a>
25 <div class="collapse bg-success" id="collapseDiv1">
26   Voilà le contenu replié et repliable pour le lien !
27 </div>
```

[Lien déclencheur](#)

Voilà le contenu replié et repliable pour le lien !

<http://getbootstrap.com/docs/5.0/components/collapse/>

# Repli (collapse) - tutoriel



<http://www.youtube.com/watch?v=fG0ThjKqzbw>



# Aussi disponible avec Bootstrap

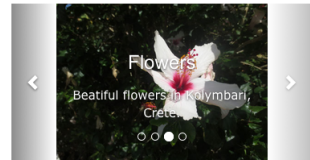
- ▶ Groupes de boutons/champs de saisie
- ▶ Jumbotron
- ▶ Barre de progression
- ▶ Pagination
- ▶ Carrousel d'images
- ▶ Scrollspy
- ▶ Media (répétition de sous-composants)
- ▶ ...



## Hello, world!

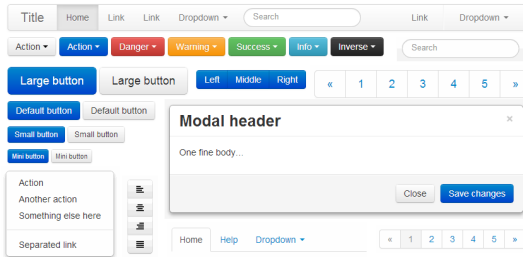
This is a simple hero unit, a simple jumbotron-style component for calling extra attention to featured content or information.

It uses utility classes for typography and spacing to space content out within the larger container.

[Learn more](#)

# En résumé

- ▶ Bootstrap propose des composants fréquemment utilisés (e.g., panels, barres de menus, alertes)
- ▶ Bootstrap stylise ces composants, mais leur logique est à programmer (e.g., fil d'Ariane, pagination)
- ▶ D'autres composants disponibles avec des extensions



# Bilan

- ▶ Bootstrap, un framework CSS avec un système de grille, des styles pré-définis et des composants supplémentaires
- ▶ Certains composants incluent des événements, dont la programmation est cachée (e.g., tooltip, carrousel)

## Prochain cours : la programmation d'événements en JavaScript

