

# HTML



# CSS



# HTML



# CSS

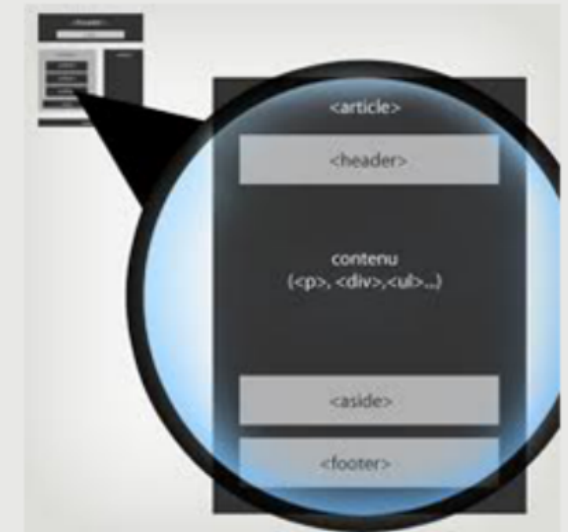
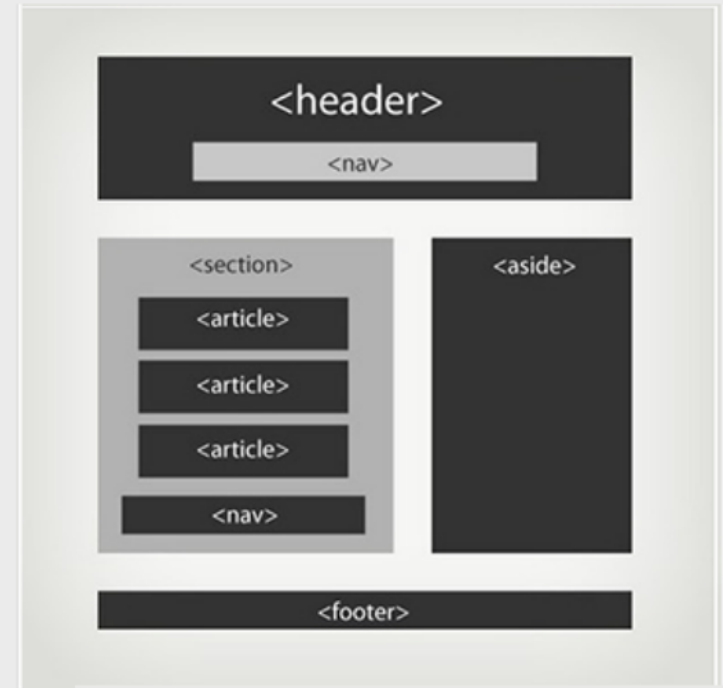


# ***Le HTML 5***

- HTML : **Hypertext Markup Language**. Format de données pour présenter des pages Web au format html.
- HTML 5 est la dernière version majeure, finalisée par le W3C le 28 Octobre 2014.
- Nouveaux éléments pour la structure sémantique des données.
- Intégration d'une couche applicative avec de nombreuses API.
- Désigne souvent un ensemble de technologies comprenant : HTML5, CSS3 et JavaScript.

# Les principales balises

- **Le doctype** : `<!doctype html>`, insensible à la casse, cette balise a été considérablement simplifiée.
- **Section** : permet de définir les grandes sections d'un document.
- **Article** : définit un texte indépendant, une portion de contenu, *ex : extrait d'un forum, article de journal ou de blog.*
- **Aside** : contenu tangentiel, c'est un élément qui se rajoute au contenu principal.
- **Header** : entête d'une section, ou d'une page.
- **Footer** : pied de page d'une section, ou d'une page.
- **Nav** : section de liens pour la navigation.
- **Figure** : désigne un conteneur pour y ajouter des images, des vidéos, ou un bloc de code, peut s'utiliser avec `<figcaption>` pour insérer une légende.



# *Les attributs*

- **Draggable** : Permet de spécifier si un élément dans la page est déplaçable.
- **Contenteditable** : Permet d'indiquer qu'une zone dans la page est éditée, l'utilisateur peut changer le contenu et manipuler le balisage.
- **Spellcheck** : Active ou non la correction orthographique d'un élément éditée.
- **Dropzone** : indique le type de contenu qui peut-être déposé dans une zone, (fonction expérimentale).
- **D'autres attributs** :
  - spécifique à certains éléments : ping sur <a> charset sur <meta>
  - Applicable à tous :  
contextmenu, hidden, data-\*, etc...

## Les balise médias

Les balises <video> et <audio> permettent d'afficher du contenu multimédia dans une page Html, ce qui peut-être une alternative au plugin Flash D'Adobe.

### Attributs :

- controls : affiche les les controle du lecteur.
- preload : (auto/none) indique si la video doit-être préchargé avec la page.
- autoplay : (true/false) démarrage auto du contenu.
- loop : lecture en boucle de la vidéo.

Pour la video :

- width : indique la largeur en px de la zone où est affché la vidéo.
- height : indique la hauteur en px de la zone où est affiché la vidéo.
- poster : indique l'URL d'une image représentant la vidéo.

### Exemple de syntaxe :

```
<video controls src="video.ogv">Ici la description alternative</video>
<video width="640" height="480" controls>
  <source src="video.webm" type="video/webm">
  <source src="video.ogv" type="video/ogg">
  <source src="video.mp4" type="video/mp4">
  <source src="video.3gp" type="video/3gp">
  Avertissement HTML5 : Veuillez mettre à jour votre navigateur pour accéder à ce contenu.
</video>
<audio controls>
  <source src="music_sample.ogg" type="audio/ogg">
  Avertissement HTML5 : Veuillez mettre à jour votre navigateur pour accéder à ce contenu.
</audio>
```

### Les Formats supportés :

Plusieurs formats de codec sont utilisables avec HTML5: H264/MP4, OGG/Theora et WebM/VP8 pour la video. MP3, AAC, OGG et WAV pour les fichiers audio.

**Remarque :** Tous les navigateurs ne sont pas compatibles entièrement avec certains codecs, pour



# Les balise médias

Les balises <video> et <audio> permettent d'afficher du contenu multimédia dans une page Html, ce qui peut-être une alternative au plugin Flash D'Adobe.

## Attributs :

- controls : affiche les les controle du lecteur.
- preload : (auto/none) indique si la video doit-être préchargé avec la page.
- autoplay : (true/false) démarrage auto du contenu.
- loop : lecture en boucle de la vidéo.

Pour la video :

- width : indique la largeur en px de la zone où est affché la vidéo.
- height : indique la hauteur en px de la zone où est affiché la vidéo.
- poster : indique l'URL d'une image représentant la vidéo.

## Exemple de syntaxe :

```
<video controls src="video.ogv">Ici la description alternative</video>
<video width="640" height="480" controls>
  <source src="video.webm" type="video/webm">
  <source src="video.ogv" type="video/ogg">
  <source src="video.mp4" type="video/mp4">
  <source src="video.3gp" type="video/3gp">
  Avertissement HTML5 : Veuillez mettre à jour votre navigateur pour accéder à ce contenu.
</video>
<audio controls>
  <source src="music_sample.ogg" type="audio/ogg">
  Avertissement HTML5 : Veuillez mettre à jour votre navigateur pour accéder à ce contenu.
</audio>
```

## Les Formats supportés :

Plusieurs formats de codec sont utilisables avec HTML5: H264/MP4, OGG/Theora et WebM/VP8 pour la video. MP3, AAC, OGG et WAV pour les fichiers audio.

**Remarque** : Tous les navigateurs ne sont pas compatibles nativement avec certains codecs, pour s'assurer une comptabilité maximal avec tous les navigateurs, il sera indispensable d'intégrer plusieurs formats.



# API HTML5

HTML5 intègre nativement des API JavaScript qui sont implémentées dans les navigateurs.

JavaScript est mis en avant dans HTML5 afin d'arriver à un niveau de performance qui se voudrait égale à des applications de bureau.

## Quelques API natives HTML5 :

- **Sélecteur CSS** : permet de retrouver facilement un, ou plusieurs élément CSS.
- **Timers** : Définit des temps d'exécution de callback. Permet notamment de rafraichir certains éléments d'une page.
- **Web Workers et Web Messaging** : Permet le traitement parallèle du programme JS principale.
- **Offline Web Application** : Permet à des applications Web de fonctionner sans Internet. Repose sur une gestion fine du cache des ressources.
- **Geolocalisation** : Permet la géolocalisation en temps réel de l'utilisateur (longitude, latitude et altitude).
- **WebStorage** : Un mode de stockage de données pour les navigateurs, (quelques Ko contre plusieurs Mo pour Web Storage).
- **DnD**: Permet de définir un élément pour recevoir des objets déposés par l'utilisateur.
- **ContentEditable** : Édition en ligne du contenu de l'élément.



# HTML



# CSS



# CSS 3

Les feuilles de style en cascades, généralement appelé CSS en anglais. Il s'agit d'un langage introduit dès les années 1990. CSS décrit la présentation de documents HTML.

## Objectifs de CSS :

- Séparer la structure d'un document de ses styles et représentation.
- Décliner les styles de représentation en fonction du média.
- Permettre la cascade de styles : différents modes de cascades peuvent se combiner, par origine, par média, par combinaisons d'éléments.

**CSS 3** : son développement commence dès 1999, parallèlement au CSS2.1, il en reprend l'ensemble des fonctionnalités.

Il intègre aussi un ensemble de nouveaux sélecteurs, il implémente les médias queries pour s'adapter aux appareils, ainsi que l'implémentation d'effets visuels poussés tels que les arrière-plans multiples et les dégradés, les transitions, l'ajout de police non-standards, les effets de textes, le multicolonnage, la gestion de la transparence, les bords arrondis, etc....

## *Quelques sélecteurs CSS3 :*

Sélection par position des fils :

**:nth-child(expression)**

Dernier enfant d'un élément parent :

**:nth-last-child(expression)**

Premier enfant d'un élément :

**:nth-first-child(expression)**

Sélection d'éléments dont l'attribut comme exactement par la valeur "stringValue" :

**[attribut^="stringValue"]**

Sélection d'éléments dont l'attribut finit exactement par la valeur "stringValue" :

**[attribut\$="stringValue"]**

Sélection d'éléments dont l'attribut comporte au moins une fois la valeur "stringValue" :

**[attribut\*="stringValue"]**

Exclusion d'éléments :

**:not([expression])**

# Les effets CSS3

**Bordures arrondies :**  
border-radius : 5px



**Ombre portée :**  
(horizontale, verticale, fondu, couleur)  
box-shadow: 5px 5px 3px gray;



**Ombre portée sur du texte :**  
text-shadow : 5px 5px 2px black;

**Css3 Text Shadow**

**Dégradé de couleurs :**  
gradient : linear-gradient(to bottom,  
rgba(102,169,221,1) 0%,rgba(10,87,150,1) 100%);



**Transition :** Passe un élément d'un état à un autre avec une transition animée.  
transition: all .8s .3s;



**Transparence :**  
opacity : 0.5



# Les Media Queries

L'attribut **media** était déjà utilisé avec CSS2 et HTML4, notamment pour modifier la mise en page d'un document pour une impression.

Aujourd'hui CSS3 offre un panel de critères qui permet d'adapter les pages en fonction de l'écran de l'utilisateur.

Le but est de cibler plus finement les périphériques de destination en fonction de leur capacité intrinsèque. On appelle cette technique le design adaptatif ou en anglais le responsive web design.

Ici on cible les écrans inférieurs à 640 pixels :

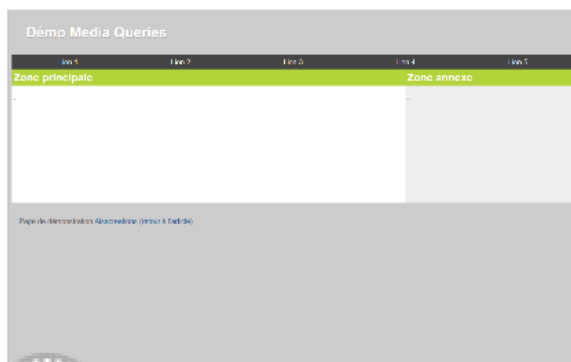
```
@media screen and (max-width: 640px) {  
  .bloc {  
    display:block;  
    clear:both;  
  }  
}
```

Ici l'on s'adresse à un écran dont la résolution en largeur est comprise entre 200 et 640 pixels :

```
@media screen and (min-width: 200px) and (max-width: 640px) {  
  .bloc {  
    display:block;  
    clear:both;  
  }  
}
```

## Exemple de Media Queries :

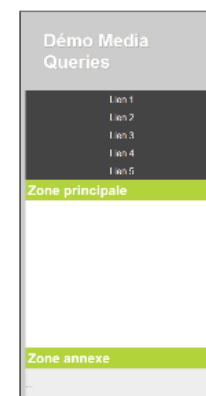
Ecran de PC : 1280 x 1024 px



Tablette : 800 x 1280 px



Smartphone : 320 x 480 px



# HTML



# CSS

