

HTML5

-

CSS3

HTML



CSS



Copyright

Copyright Emmanuel Fernandez Orvault Tous droits réservés.

Toute représentation ou reproduction, intégrale ou partielle, faite sans le consentement de l'auteur, de ses ayants droit ou ayants cause, est illicite (loi du 11 mars 1957, alinéa 1er de l'article 40). Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait une contrefaçon sanctionnée par les articles 425 et suivants du Code pénal. La loi du 11 mars 1957 autorise uniquement, aux termes des alinéas 2 et 3 de l'article 41, les copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective d'une part et, d'autre part, les analyses et les courtes citations dans un but d'exemple et d'illustration.

Plan

La page de base

Les nouvelles balises HTML5

Les nouveaux sélecteurs et propriétés CSS3

Les coins arrondis

Les ombres

Couleurs et transparence

Dégradé linéaire

Insertion de fontes

Les balises audio et video

Les nouveautés de la balise input

Autres nouveautés de formulaire

Transformations, transitions et animations

Les Media Queries

Les nouvelles API du HTML5

La page de base

- Document XHTML 4 Strict

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Document sans nom</title>
</head>

<body>
</body>
</html>
```

- Document HTML5

```
<!doctype html>
<html>
<head>
<meta charset="UTF-8">
<title>Document sans nom</title>
</head>

<body>
</body>
</html>
```

La page de base

- Simplification
 - Le doctype
 - Il a été allégé (plus de numéro de version)
 - Permet de préciser quel est le type de document qui va suivre, afin de permettre au navigateur de savoir quel langage il devra interpréter.
 - Le <html>
 - Elément racine du document (« root » en anglais)
 - Ne peut pas être contenu dans un autre élément.
 - Contient les balises <head> et <body>

La page de base

- Simplification
 - Le <head>
 - La balise contient plusieurs types d'éléments :
 - L'encodage avec la balise **meta** ou **charset**.
 - Le titre de la page.
 - Les liens avec la balise **link**.
 - Et d'autres indications par des métas.
 - Encodage :
 - Simplifié avec uniquement l'attribut **charset** dans la balise **meta**

La page de base

- Simplification

- Retour de la permissivité du HTML 4

- Insensible à la casse : possibilité d'écrire les balises en minuscule ou majuscule
 - Les attributs ne sont plus obligatoirement valorisés
 - La valeur des attributs n'est plus obligatoirement comprise entre guillemets
 - Les balises vides ne sont plus obligatoirement fermées

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="UTF-8" >
    <title>Document sans nom</title>
  </head>

  <body>
    <FORM>
      CheckBox : <INPUT type="checkbox" checked>
    </form>
  </body>
</html>
```

La page de base

- Simplification

- Liaison vers un fichier CSS

- L'attribut type n'est plus obligatoire :

```
<link href="../WSCSS/chap3.css" rel="stylesheet" type="text/css">
```



```
<link href="../WSCSS/chap3.css" rel="stylesheet" >
```

- Liaison vers un fichier JavaScript

- L'attribut type n'est plus obligatoire :

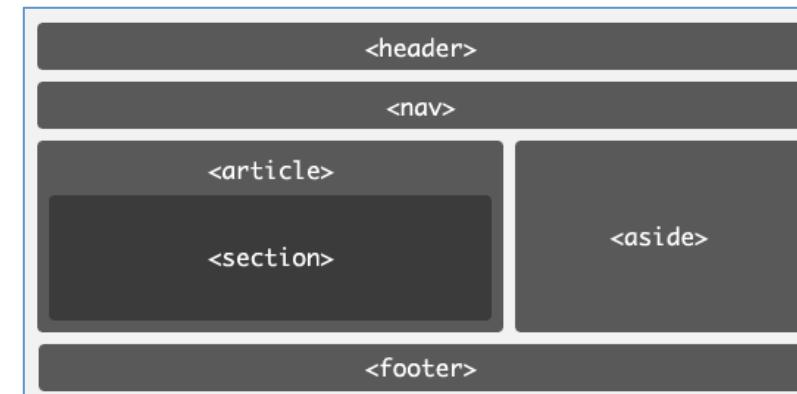
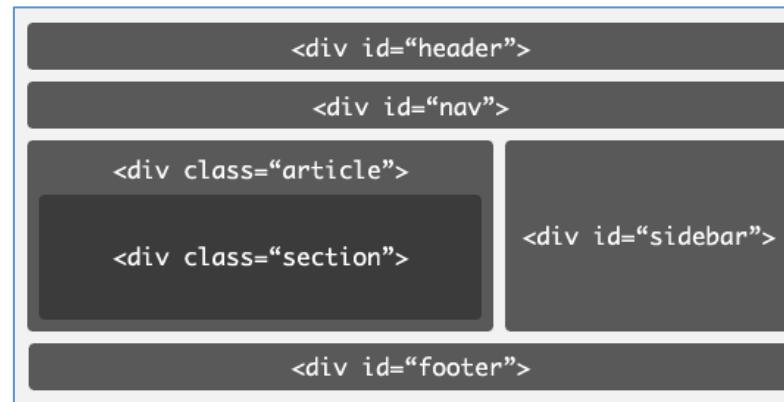
```
<script type="text/javascript" src="scripts.js"></script>
```



```
<script src="scripts.js"></script>
```

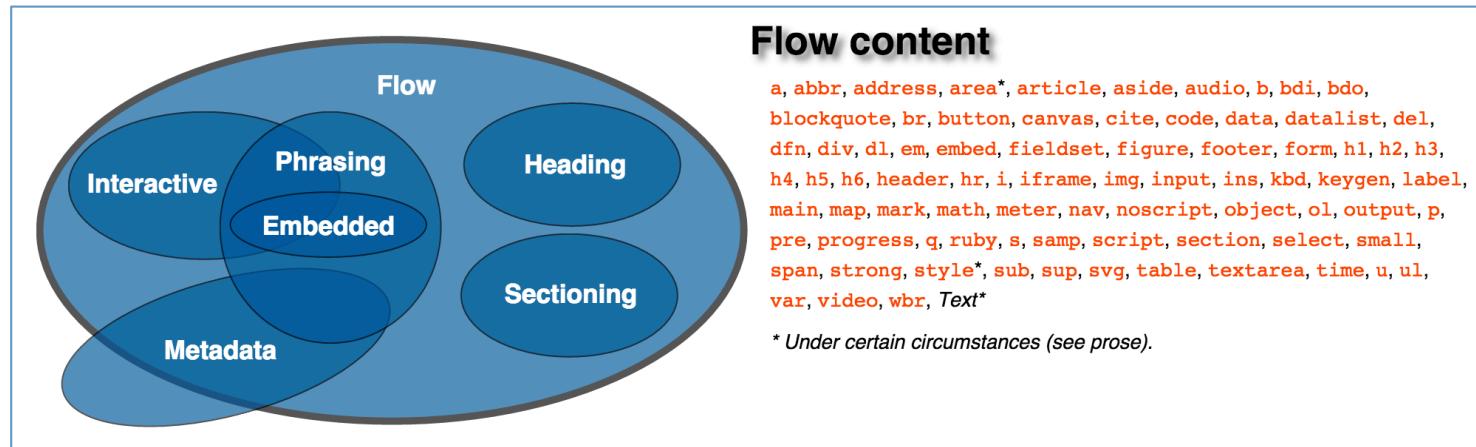
Les nouvelles balises HTML5

- Différences entre HTML 4 et HTML5



Les nouvelles balises HTML5

- Les éléments HTML5 sont regroupés en 7 sections
<http://www.w3.org/TR/html5/dom.html#content-models>



Les nouvelles balises HTML5

- Les éléments HTML5 sont regroupés en 7 sections
 - Flow Content :
 - Eléments contenus dans le body
 - Metadata content :
 - Mettant en relation des documents entre eux
 - base link meta noscript script style template title
 - Sectioning content:
 - Eléments qui créent des sections dans la page
 - article aside nav section
 - Heading content
 - Eléments représentant les en-tête d'une section
 - h1 h2 h3 h4 h5 h6
 - Phrasing content :
 - Eléments qui apparaissent dans le texte (inline)
 - audio b br button canvas datalist em embed i
 - Embedding content
 - Eléments permettant d'ajouter/afficher des ressources externes
 - audio canvas embed iframe img math object svg video
 - Interactive content :
 - Eléments ayant une interaction avec l'utilisateur
 - button embed iframe img input ...

Les nouvelles balises HTML5

- Gestion des sections
 - L'élément `<section>` permet de définir les grandes sections d'un document (les chapitres, les en-tête et pied-de-page, autres sections, ...). Il peut être combiné avec les éléments h1 à h6 pour une meilleure définition de la structure du document.

```
<section>
  <h1>Section Principale</h1>
  <section>
    <h1>Introduction</h1>
    <p>Sed tamen haec cum ita tutius observentur.</p>
  </section>
  <section>
    <h1>Conclusion</h1>
    <p>Has autem provincias, quas Orontes ambien.</p>
  </section>
</section>
```

Les nouvelles balises HTML5

- Gestion des sections
 - L'élément `<nav>` permet de définir une section de navigation avec des liens internes au document.

```
<nav>
  <ul>
    <li><a href="index.html">Page d'accueil</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</nav>
```

Les nouvelles balises HTML5

- Gestion des sections
 - L'élément `<aside>` est une section dont le contenu est un complément par rapport à ce qui l'entoure, qui n'est pas forcément en lien direct avec le contenu mais qui peut apporter des informations supplémentaires.

```
<aside>
  <h4>Références à l'auteur</h4>
  <nav>
    <ul>
      <li><a href="#">Bibliographie</a></li>
      <li><a href="#">Vie publique</a></li>
      <li><a href="#">...</a></li>
    </ul>
  </nav>
</aside>
```

Les nouvelles balises HTML5

- Gestion des sections
 - L'élément `<article>` représente un texte *indépendant*, une portion de contenu, comme par exemple un article de journal, de blog ou de forum.
 - Il vise à baliser des blocs de contenu utiles que l'on pourrait extraire du document tout en conservant leur sens et leurs informations.

```
<section>
  <h1>Section Principale</h1>
  <article>
    <h1>Nouveautés HTML5</h1>
    <p>Sed tamen haec cum ita tutius observentur.</p>
  </article>
  <article>
    <h1>Nouveautés Java EE 7</h1>
    <p>Has autem provincias, quas Orontes ambien.</p>
  </article>
</section>
```

Les nouvelles balises HTML5

- En-tête et pied de page :
 - L'élément `<header>` représente l'en-tête d'une section (section, article...) ou d'une page entière. Il peut contenir un logo, un ou plusieurs titres, d'autres informations d'introduction, une navigation, un formulaire de recherche, ...

```
<article>
  <header>
    <h1>Titre de l'article</h1>
    <p>Date : 21/07/2015</p>
  </header>

  <p>Quam ob rem cave Catoni anteponas ne istum quidem</p>
</article>
```

Les nouvelles balises HTML5

- En-tête et pied de page :
 - L'élément `<footer>` représente le pied de page, ou bien la conclusion d'une section.
 - S'y trouvent des informations concernant l'auteur, des mentions légales, une navigation ou une pagination, un logo de rappel, des coordonnées, des dates de publication, ...

```
<article>
  <p>Quam ob rem cave Catoni anteponas ne istum quidem</p>

  <footer>
    <p>Copyright EFE, le
      <time datetim="2015-07-15">15 juillet 2015</time>
    </p>
  </footer>
</article>
```

Les nouveaux sélecteurs et propriétés CSS3

- Documentation :
 - <http://www.w3.org/TR/selectors/>

Pattern	Meaning	Described in section	First defined in CSS level
*	any element	Universal selector	2
E	an element of type E	Type selector	1
E[foo]	an E element with a "foo" attribute	Attribute selectors	2
E[foo="bar"]	an E element whose "foo" attribute value is exactly equal to "bar"	Attribute selectors	2
E[foo~="bar"]	an E element whose "foo" attribute value is a list of whitespace-separated values, one of which is exactly equal to "bar"	Attribute selectors	2
E[foo^="bar"]	an E element whose "foo" attribute value begins exactly with the string "bar"	Attribute selectors	3
E[foo\$="bar"]	an E element whose "foo" attribute value ends exactly with the string "bar"	Attribute selectors	3
E[foo*="bar"]	an E element whose "foo" attribute value contains the substring "bar"	Attribute selectors	3
E[foo=“en”]	an E element whose "foo" attribute has a hyphen-separated list of values beginning (from the left) with "en"	Attribute selectors	2
E:root	an E element, root of the document	Structural pseudo-classes	3
E:nth-child(n)	an E element, the n-th child of its parent	Structural pseudo-classes	3
E:nth-last-child(n)	an E element, the n-th child of its parent, counting from the last one	Structural pseudo-classes	3
E:nth-of-type(n)	an E element, the n-th sibling of its type	Structural pseudo-classes	3
E:nth-last-of-type(n)	an E element, the n-th sibling of its type, counting from the last one	Structural pseudo-classes	3
E:first-child	an E element, first child of its parent	Structural pseudo-classes	2
E:last-child	an E element, last child of its parent	Structural pseudo-classes	3
E:first-of-type	an E element, first sibling of its type	Structural pseudo-classes	3
E:last-of-type	an E element, last sibling of its type	Structural pseudo-classes	3
E:only-child	an E element, only child of its parent	Structural pseudo-classes	3
E:only-of-type	an E element, only sibling of its type	Structural pseudo-classes	3
E:empty	an E element that has no children (including text nodes)	Structural pseudo-classes	3

Les nouveaux sélecteurs et propriétés CSS3

- Les sélecteurs d'attribut :
 - Attribut commençant par :

```
<p couleur="bleu">Paragraphe bleu</p>
<p couleur="bleuClair">Paragraphe bleu clair</p>
<p couleur="bleuFonce">Paragraphe bleu fonce</p>
<p couleur="vert">Paragraphe vert</p>
<p couleur="vertFonce">Paragraphe vert fonce</p>
```

```
p[couleur^="bleu"]{
    color:blue;
}

p[couleur^="vert"]{
    color:green;
}
```



Les nouveaux sélecteurs et propriétés CSS3

- Les sélecteurs d'attribut :
 - Attribut finissant par :

```
p[couleur$="Clair"]{  
    font-weight:bold;  
}  
  
p[couleur$="Fonce"]{  
    font-style:italic;  
}
```



- Attribut contenant :

```
p[couleur*="rtFonc"]{  
    color:rgba(32,82,41,1.00);  
    font-weight:bold;  
}  
  
p[couleur*="euFonc"]{  
    color:rgba(30,47,166,1.00);  
    font-weight:bold;  
}
```



Les nouveaux sélecteurs et propriétés CSS3

- Pseudo-classes de structure :
 - En fonction de son **numéro d'ordre** :

```
p:nth-child(2){  
    font-style:italic;  
}  
  
p:nth-child(3){  
    font-weight:bold;  
}
```



Les nouveaux sélecteurs et propriétés CSS3

- Pseudo-classes de structure :
 - First Child versus First of Type
 - E:first-child : Tous les éléments E premiers fils du parent
 - E:first-of-type : Tous les éléments E premiers fils E du parent
 - Last Child versus Last of Type
 - E:last-child : Tous les éléments E derniers fils du parent
 - E:last-of-type : Tous les éléments E derniers fils E du parent

Les nouveaux sélecteurs et propriétés CSS3

- Pseudo-classes de structure :
 - Exemple

```
<body>
  <p couleur="bleu">Paragraphe bleu</p>
  <p couleur="rouge">Paragraphe rouge</p>
  <p couleur="vert">Paragraphe vert</p>
  <div>
    <p couleur="bleu">Paragraphe bleu</p>
    <p couleur="rouge">Paragraphe rouge</p>
    <p couleur="vert">Paragraphe vert</p>
  </div>
</body>
```

```
p:first-of-type{
  font-style:italic;
}

p:last-of-type{
  font-weight:bold;
}

p:first-child{
  color:blue;
}

p:last-child{
  color:green;
}
```

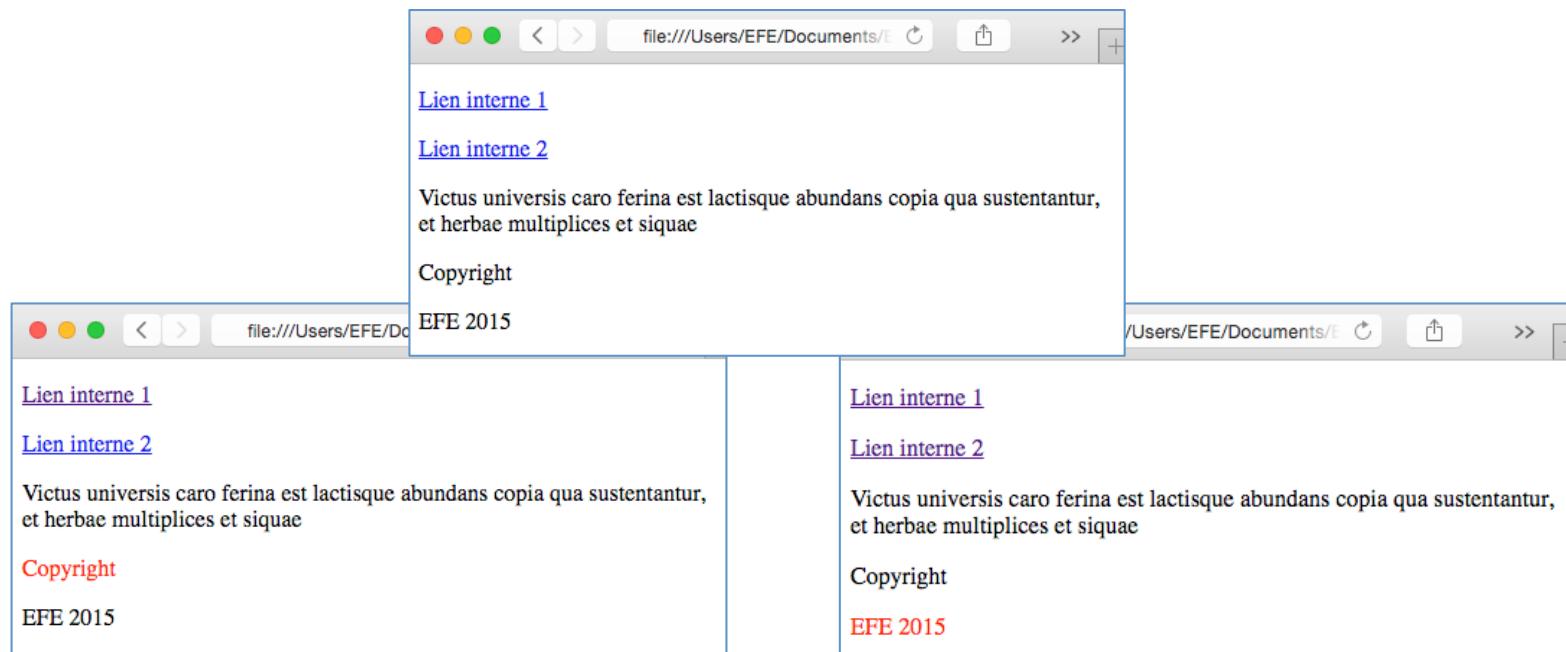


Les nouveaux sélecteurs et propriétés CSS3

- Pseudo-classes de cible :

```
<body>
  <p><a href="#footer1">Lien interne 1</a></p>
  <p><a href="#footer2">Lien interne 2</a></p>
  <p>Victus universis caro ferina est lactisque abundans copia qua
sustentantur, et herbae multiplices et siqueae</p>
  <footer>
    <p id="footer1">Copyright</p>
    <p id="footer2">EFE 2015</p>
  </footer>
</body>
```

```
p:target{
  color:red;
}
```



Les nouveaux sélecteurs et propriétés CSS3

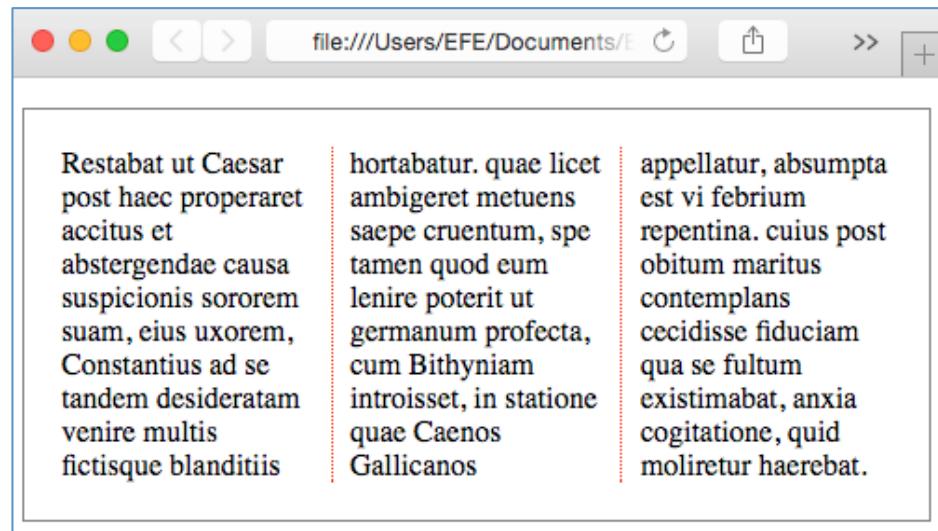
- Les propriétés colonnes :
 - Permet de définir le nombre de colonnes
 - column-count
 - Nombre de colonnes à afficher
 - column-gap
 - Espace entre 2 colonnes
 - column-rule
 - Séparateur entre deux colonnes

```
p{  
    column-count:3;  
    column-gap:20px;  
    column-rule:1px red dotted;  
}
```

Les nouveaux sélecteurs et propriétés CSS3

- Les propriétés colonnes :
 - Problème : tous les navigateurs n'implémentent pas cette fonctionnalité.
 - Utilisation d'une propriété spécifique

```
p{  
    padding:20px ;  
    border : 1px solid grey;  
  
    -webkit-column-count:3;  
    -webkit-column-gap:20px;  
    -webkit-column-rule:1px red dotted;  
  
    -moz-column-count:3;  
    -moz-column-gap:20px;  
    -moz-column-rule:1px red dotted;  
  
    column-count:3;  
    column-gap:20px;  
    column-rule:1px red dotted;  
}
```



Les nouveaux sélecteurs et propriétés CSS3

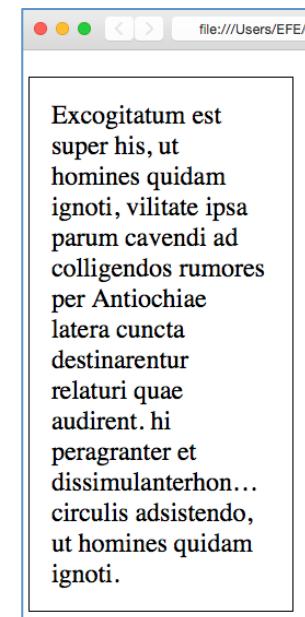
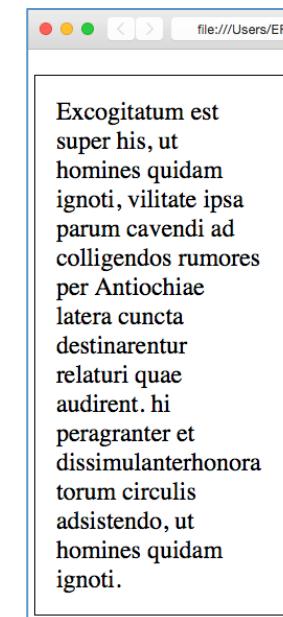
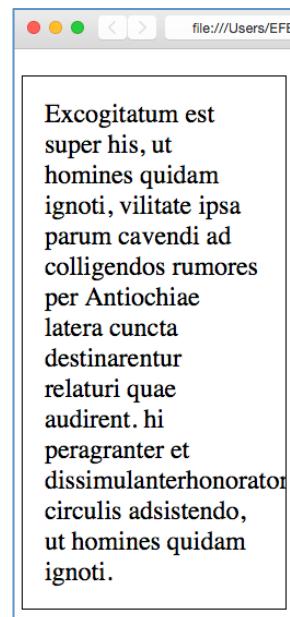
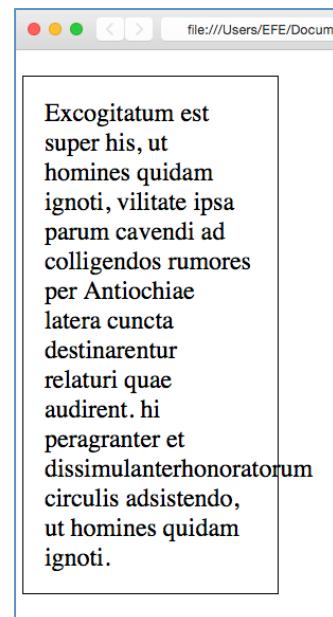
- Les débordements :

```
p{  
    font-size:1.5em;  
    border:1px solid black;  
    padding:20px;  
    width:200px;  
}
```

```
p{  
    font-size:1.5em;  
    border:1px solid black;  
    padding:20px;  
    width:200px;  
    overflow: hidden;  
}
```

```
p{  
    font-size:1.5em;  
    border:1px solid black;  
    padding:20px;  
    width:200px;  
    word-wrap: break-word;  
}
```

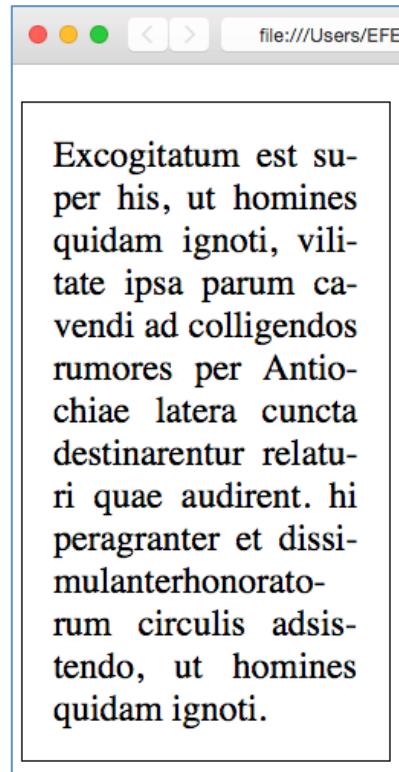
```
p{  
    font-size:1.5em;  
    border:1px solid black;  
    padding:20px;  
    width:200px;  
    overflow: hidden;  
    text-overflow:ellipsis;  
}
```



Les nouveaux sélecteurs et propriétés CSS3

- Les débordements :

```
p{  
    text-align:justify;  
    font-size:1.5em;  
    border:1px solid black;  
    padding:20px;  
    width:200px;  
    overflow: hidden;  
  
    -webkit-hyphens: auto;  
    -moz-hyphens: auto;  
    -ms-hyphens: auto;  
    -o-hyphens: auto;  
    hyphens:auto;  
}
```



```
p{  
    font-size:1.5em;  
    border:1px solid black;  
    padding:20px;  
    width:400px;  
    white-space:nowrap;  
    overflow: hidden;  
    text-overflow:ellipsis;  
}
```

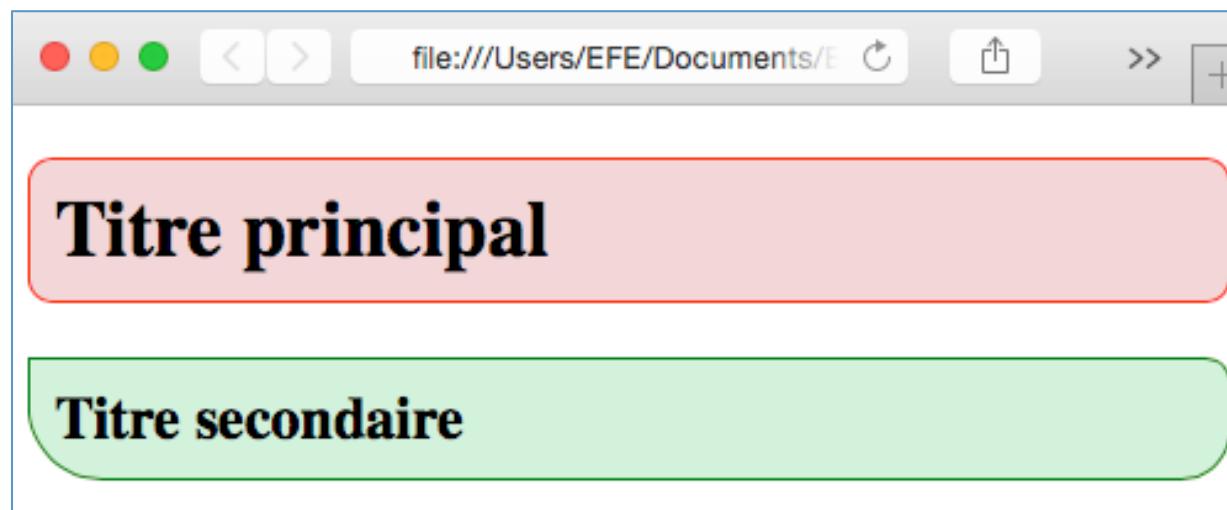


Les coins arrondis

- border-radius

```
h1 {  
    border : 1px red solid;  
    background-color:rgba(243,215,216,1.00);  
    padding: 10px;  
    border-radius: 10px;  
}
```

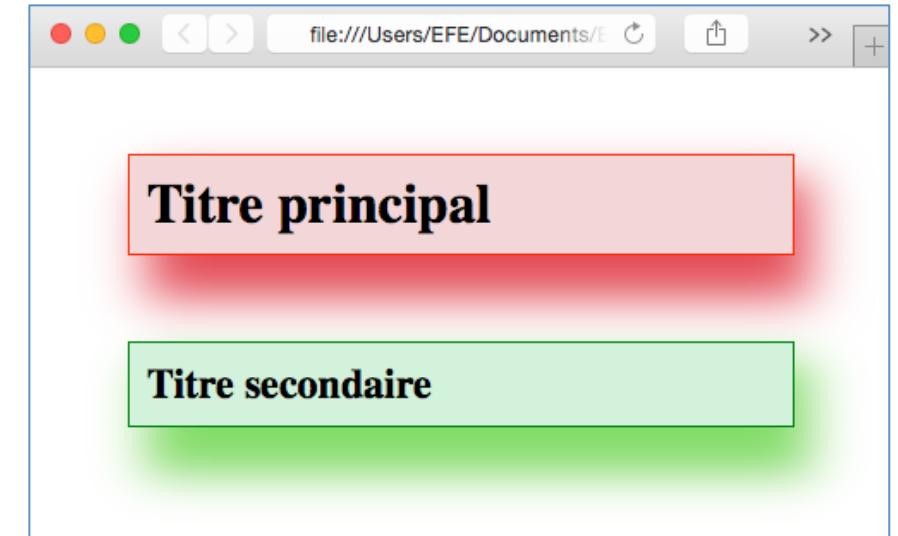
```
h2 {  
    border : 1px green solid;  
    background-color:rgba(211,242,220,1.00);  
    padding: 10px;  
    border-radius: 0px 10px 20px 30px ;  
}
```



Les ombres

- Les ombres sur les boîtes
 - Propriété **box-shadow** :
 - Décalage horizontal de l'ombre
 - Décalage vertical de l'ombre
 - Nombre de pixels flous
 - Couleur de l'ombre

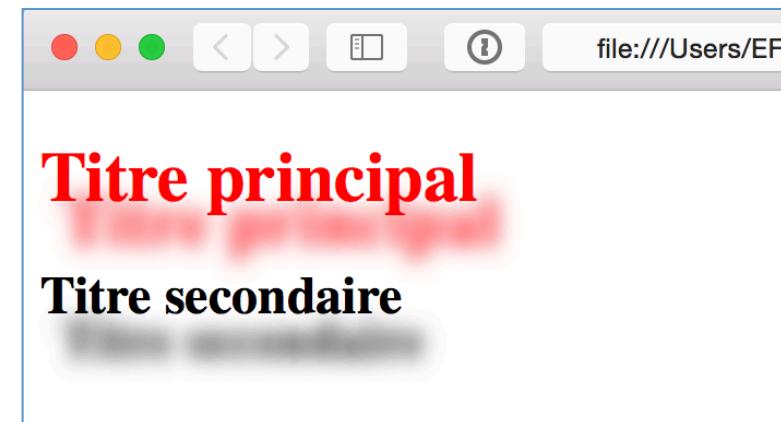
```
h1 {  
    border : 1px red solid;  
    background-color:rgba(243,215,216,1.00);  
    padding: 10px;  
    margin: 50px;  
    box-shadow: 10px 20px 30px rgba(219,26,30,1.00);  
}  
  
h2 {  
    border : 1px green solid;  
    background-color:rgba(211,242,220,1.00);  
    padding: 10px;  
    margin: 50px;  
    box-shadow: 10px 20px 30px rgba(51,205,3,1.00)  
}
```



Les ombres

- Les ombres sur les textes
 - Propriété **text-shadow** :
 - Décalage horizontal de l'ombre
 - Décalage vertical de l'ombre
 - Nombre de pixels flous
 - Couleur de l'ombre

```
h1 {  
    color:red;  
    text-shadow: 10px 20px 10px #ff0000;  
}  
  
h2 {  
    text-shadow: 10px 20px 10px #000000  
}
```



Couleurs et transparence

- **RGBa**

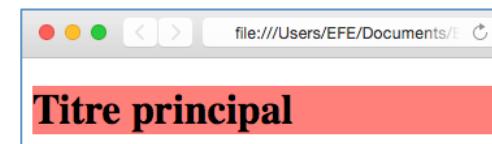
rgba (rouge, vert, bleu, transparence)

(avec **transparence** compris entre 0 et 1)

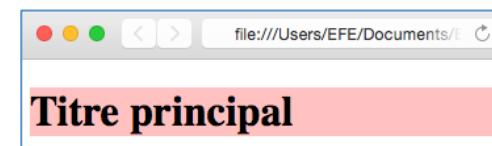
```
h1 {  
    background-color:rgba(255,0,0,1);  
}
```



```
h1 {  
    background-color:rgba(255,0,0,0.6);  
}
```



```
h1 {  
    background-color:rgba(255,0,0,0.3);  
}
```



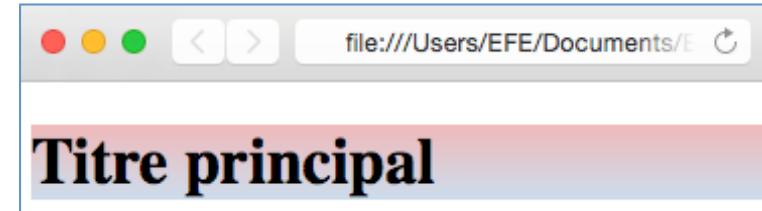
```
h1 {  
    background-color:rgba(255,0,0,0);  
}
```



Dégradé linéaire

- **linear-gradient**
 - La fonction **linear-gradient** permet de créer un dégradé linéaire. Elle prend 3 valeurs, dont 2 sont obligatoires :
 - direction du dégradé (optionnel)
 - Par défaut du haut vers le bas ("to bottom")
 - couleur de départ
 - couleur de fin

```
h1{  
    background: linear-gradient(#F0BABA, #CDDEED);  
}
```



Dégradé linéaire

- linear-gradient

```
h1{  
    background:linear-gradient(to top, #F0BABA, #CDDEED);  
}
```

Titre principal

```
h1{  
    background:linear-gradient(to left, #F0BABA, #CDDEED);  
}
```

Titre principal

```
h1{  
    background:linear-gradient(to top right, #F0BABA, #CDDEED);  
}
```

Titre principal

```
h1{  
    background:linear-gradient(to right, #FF0000 30%, #00FF00 70%);  
}
```



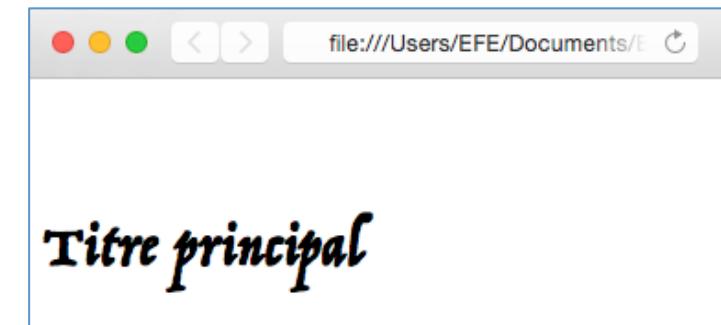
```
h1{  
    background:linear-gradient(to right, red,  
                                orange,yellow,green,blue,indigo,violet);  
}
```



Insertion de fontes

- **@font-face**
 - Permet d'insérer des fontes personnalisées à des fins d'utilisation dans des pages HTML
 - Exemple simple :

```
@font-face {  
    font-family: 'trattatelloregular';  
    src: url('trattatello/trattatello-webfont.ttf');  
}  
  
h1{  
    font-family:'trattatelloregular';  
}
```



Insertion de fontes

- **@font-face**
 - **Problème** : le navigateur doit supporter le format **TrueType**, ce qui n'est pas toujours le cas (ex : IE)
 - Donc besoin de proposer différents format de police afin que tous les navigateurs puissent intégrer la nouvelle police

```
@font-face {  
    font-family: 'trattatelloregular';  
    src: url('trattatello/trattatello-webfont.eot'),  
        url('trattatello/trattatello-webfont.eot?#iefix') format('embedded-opentype'),  
        url('trattatello/trattatello-webfont.woff2') format('woff2'),  
        url('trattatello/trattatello-webfont.woff') format('woff'),  
        url('trattatello/trattatello-webfont.ttf') format('truetype'),  
        url('trattatello/trattatello-webfont.svg#trattatelloregular') format('svg');  
    font-weight: normal;  
    font-style: normal;  
}
```

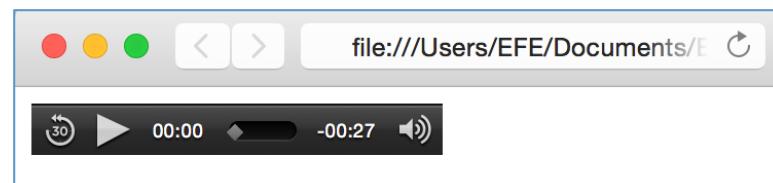
Les balises audio et video

- Audio :
 - Problèmes de compatibilité entre les différents navigateurs :

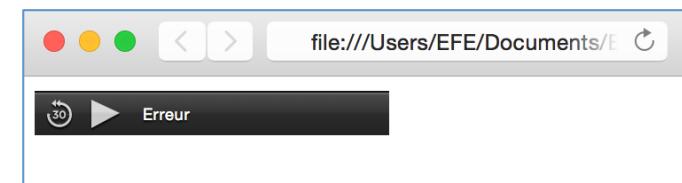
	V 14 et +	V 20 et +	V. 9	V 5.1 et +	12 et +
mp3	✓	✓	✓	✓	✗
ogg	✓	✓	✗	✗	✓
wav	✓	✓	✗	✓	✓

– Exemple avec Safari :

mp3



ogg



Les balises audio et video

- Audio :
 - Balise simple

```
<audio src="audio/ghostwriter.mp3" controls> </audio>
```

- **src** : indique le fichier audio
- **controls** : affiche une barre de commandes

- Balise avancée :

```
<audio controls>
  <source src="audio/ghostwriter.ogg" />
  <source src="audio/ghostwriter.mp3" />
</audio>
```

Les balises audio et video

- Audio :
 - Les attributs :
 - **controls** : affiche la barre de commande
 - **autoplay** : lance la musique automatiquement au chargement de la page
 - **loop** : permet de jouer la musique en boucle
 - **preload** : indique au navigateur de charger le fichier au chargement de la page et non au premier appel (ne devrait pas être utilisé en même temps que autoplay)

```
<audio controls autoplay loop preload="true">
    <source src="audio/ghostwriter.ogg" />
    <source src="audio/ghostwriter.mp3" />
</audio>
```

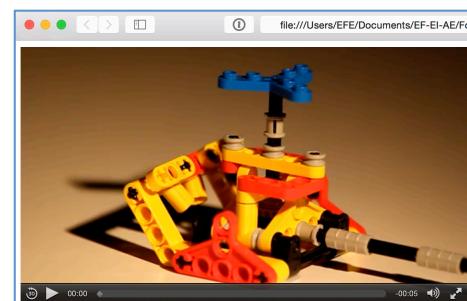
Les balises audio et video

- Vidéo
 - Problèmes de compatibilité entre les différents navigateurs :

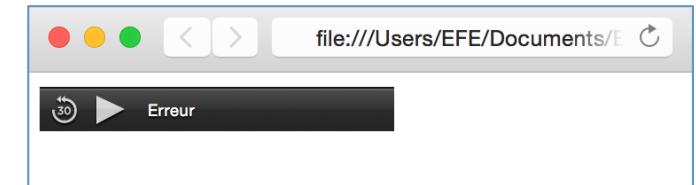
H.264/MP4	✓	✓	✓	✓	✓
OGG	✓	✓	✗	✗	✓
WEBM	✓	✓	✗	✗	✓

- Exemple avec Safari :

mp4



ogg video



Les balises audio et video

- Vidéo :
 - Balise simple

```
<video src="video/small.mp4" controls></video>
```

- **src** : indique le fichier vidéo
- **controls** : affiche une barre de commandes

- Balise avancée :

```
<video controls>
  <source src="video/small.webm" />
  <source src="video/small.mp4" />
  <source src="video/small.ogv" />
</video>
```

Les balises audio et video

- Vidéo :
 - Les attributs :
 - **controls** : affiche la barre de commande
 - **autoplay** : lance la vidéo automatiquement au chargement de la page
 - **loop** : permet de jouer la vidéo en boucle
 - **preload** : indique au navigateur de charger le fichier au chargement de la page et non au premier appel (ne devrait pas être utilisé en même temps que autoplay)
 - **poster** : image à afficher tant que la vidéo n'est pas disponible ou pas chargée

```
<video controls autoplay loop poster="img/image.png">
  <source src="video/small.webm" />
  <source src="video/small.mp4" />
  <source src="video/small.ogv" />
</video>
```

Les nouveautés de la balise input

- 13 nouveaux type input

- search
- email
- url
- tel
- number
- range
- date
- month
- week
- time
- datetime
- datetime-local
- color

- Problème :
- Tous les navigateurs ne sont pas compatibles !
- Dans ce cas, un champ input de type **text** est affiché

Les nouveautés de la balise input

- Compatibilité

Field types	Chrome	Firefox	Opera	Safari	IE
▶ <code>input type=text</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Partial ○
▶ <code>input type=search</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=tel</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=url</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=email</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=date</code>	Yes ✓	No ✗	Yes ✓	No ✗	No ✗
▶ <code>input type=month</code>	Yes ✓	No ✗	Yes ✓	No ✗	No ✗
▶ <code>input type=week</code>	Yes ✓	No ✗	Yes ✓	No ✗	No ✗
▶ <code>input type=time</code>	Yes ✓	No ✗	Yes ✓	No ✗	No ✗
▶ <code>input type=datetime</code>	No ✗	No ✗	No ✗	No ✗	No ✗
▶ <code>input type=datetime-local</code>	Yes ✓	No ✗	Yes ✓	No ✗	No ✗
▶ <code>input type=number</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=range</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=color</code>	Yes ✓	Yes ✓	Yes ✓	No ✗	No ✗
▶ <code>input type=checkbox</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=image</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
▶ <code>input type=file</code>	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓

Les nouveautés de la balise input

- Exemple :

```
<h1>Nouveaux Champs HTML5</h1>
<form>
    <h2>Nouveaux types de champ</h2>

    <h3>email :</h3>
    <input type="email" value="efe@test.fr" />

    <h3>date :</h3>
    <input type="date" min="2014-01-15" max="2016-01-15" value="2015-01-15" />

    <h3>range</h3>
    <input type="range" step="10" min="0" max="50" value="20" />

    <h3>search</h3>
    <input name="search" type="search" placeholder="Rechercher ..." />

    <h3>tel</h3>
    <input type="tel" placeholder="(111) 111-1111" pattern="^((\d{3}) (\d{3}) (\d{2}) (\d{2}))$" />

    <h3>couleur</h3>
    <input type="color" placeholder="ex: #12FF45" />

    <h3>nombres</h3>
    <input type="number" min="-10" max = "10" step="1" value="0"/>

    <input type="submit" value="Envoyer" />
</form>
```

Les nouveautés de la balise input

- Exemple :

Chrome

Nouveaux Champs HTML5

Nouveaux types de champ

email :

date :

range



search

tel

couleur



nombres

Safari

Nouveaux Champs HTML5

Nouveaux types de champ

email :

date :

range



search

tel

couleur

nombres

Firefox

Nouveaux Champs HTML5

Nouveaux types de champ

email :

date :

range



search

tel

couleur



nombres

Les nouveautés de la balise input

- Les nouveaux attributs
 - placeholder
 - Attribut qui permet de renseigner un texte indicatif par défaut dans un champ de formulaire.
 - C'est une valeur qui s'efface dès que l'utilisateur active le champ de formulaire (ou commence à écrire dedans)
 - autofocus
 - permet de spécifier si oui ou non la page devrait sélectionner le contrôle après son chargement.
 - Un seul élément form par document peut avoir un attribut autofocus.
 - Cet attribut ne peut pas être appliqué si l'attribut type vaut hidden.

Les nouveautés de la balise input

- Les nouveaux attributs
 - required
 - Attribut indiquant que l'utilisateur doit renseigner la valeur avant d'envoyer le formulaire.
 - Il ne peut pas être utilisé lorsque la valeur de l'attribut type est hidden, image, ou un bouton.

email :

Envoyer

 Veuillez renseigner ce champ.

Chrome

email :

Envoyer

Veuillez compléter ce champ

Firefox

Les nouveautés de la balise input

- Les nouveaux attributs
 - **autocomplete**
 - Attribut indiquant si oui ou non la valeur du contrôle peut être automatiquement complétée par le navigateur.

A screenshot of a web browser showing a search interface. At the top, the word "search" is written in bold. Below it is a search bar containing the placeholder text "Rechercher ...". To the right of the search bar is a button labeled "Envoyer". Underneath the search bar, there is a dropdown menu or suggestion list with two items: "navigateur" and "azerty". The word "navigateur" is highlighted with a blue border, indicating it is the currently selected or focused suggestion.

- Pas implémentée par tous les navigateurs

Les nouveautés de la balise input

- Les nouveaux attributs
 - pattern
 - Une expression régulière qui sert à vérifier la valeur.
 - Exemple : une lettre majuscule comprise entre A et F suivie de 5 chiffres

```
<input name="search" type="search" pattern="[A-F][0-9]{5}" />
```

A screenshot of a web browser showing a search input field. The input field has the placeholder "search". Inside the field, the text "A1234" is entered. To the right of the input field are two buttons: a red "X" button and a blue "Envoyer" button. Below the input field, a tooltip-like message box appears, containing an orange exclamation mark icon and the text "Veuillez respecter le format requis." (Please respect the required format).

Autres nouveautés de formulaire

- La balise `datalist` :
 - Elément de formulaire permettant de lier une liste de choix à un élément `input`.

```
<label for="choix_villes">Indiquez la ville :</label>  
  
<input list="villes" type="text" id="choix_villes">  
  
<datalist id="villes">  
  <option value="Nantes">  
  <option value="Bordeaux">  
  <option value="Marseille">  
  <option value="Lille">  
</datalist>
```

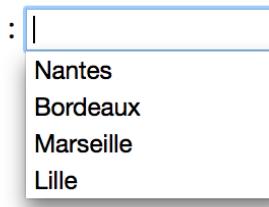
Indiquez la ville :



Nantes
Bordeaux
Marseille
Lille

Chrome

Indiquez la ville :



Nantes
Bordeaux
Marseille
Lille

Firefox

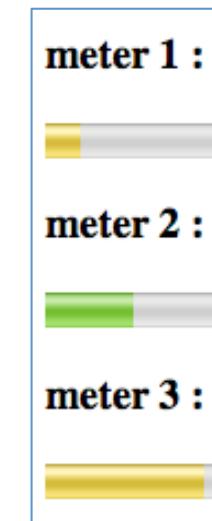
Autres nouveautés de formulaire

- La balise meter
 - Représente une jauge mesurant une valeur comprise entre un minimum et un maximum, soit dans un intervalle établi.
 - Les valeurs négatives peuvent être utilisées

```
<h3>meter 1 :</h3>
<meter min="0" max="100" value="20" low="40" high="80"></meter>

<h3>meter 2 :</h3>
<meter min="0" max="100" value="50" low="40" high="80"></meter>

<h3>meter 3 :</h3>
<meter min="0" max="100" value="90" low="40" high="80"></meter>
```



Autres nouveautés de formulaire

- La balise progress
 - Indique la progression d'une tâche au cours du temps.

```
<h3>progress :</h3>
<progress></progress>

<h3>progress avec valeur et max:</h3>
<progress value="75" max="100"></progress>
```



Autres nouveautés de formulaire

- L'attribut `contenteditable`
 - Attribut énuméré qui indique si l'élément peut être édité par l'utilisateur.
 - Si c'est le cas, le navigateur modifie son interface afin de permettre l'édition. Cet attribut doit avoir l'une des valeurs suivantes :
 - `true` ou la chaîne de caractères vide : indique que l'élément est éditable
 - `false` : indique que l'élément ne peut pas être édité.

```
<h3>Nouvelle fonctionnalité d'édition : </h3>
```

```
Ceci est une <span contenteditable="true">chaine</span>
```

Nouvelle fonctionnalité d'édition :

Ceci est une chaine

Nouvelle fonctionnalité d'édition :

Ceci est une chaine modifiée

Nouvelle fonctionnalité d'édition :

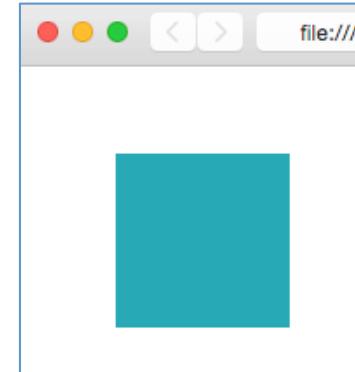
Ceci est une chaine modifiée

Transformations, transitions et animations

- Les transformations :
 - Elles peuvent être :
 - Des **translations**,
 - Des **rotations**,
 - Des **réductions ou agrandissements (zoom)**
 - Des **distorsions**
 - Exemples :

```
#bloc{  
    width:100px;  
    height:100px;  
    background-color:#27A9B6;  
    margin : 50px 0 0 50px;  
}
```

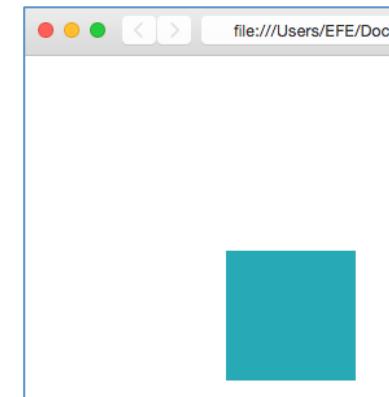
```
<body>  
    <div id="bloc" >  
    </div>  
</body>
```



Transformations, transitions et animations

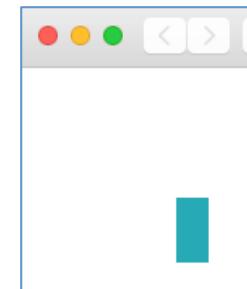
- Translation :

```
#bloc{  
    -webkit-transform:translateX(100px) translateY(100px);  
    -moz-transform:translateX(100px) translateY(100px);  
    -o-transform:translateX(100px) translateY(100px);  
    transform:translateX(100px) translateY(100px);  
}
```



- Zoom :

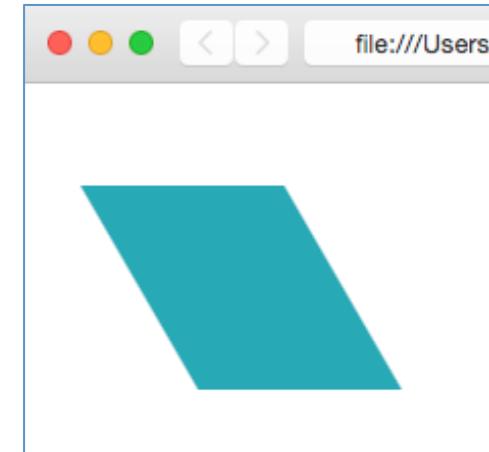
```
#bloc{  
    -webkit-transform:scale(0.2,0.4);  
    -moz-transform:scale(0.2,0.4);  
    -o-transform:scale(0.2,0.4);  
    transform:scale(0.2,0.4);  
}
```



Transformations, transitions et animations

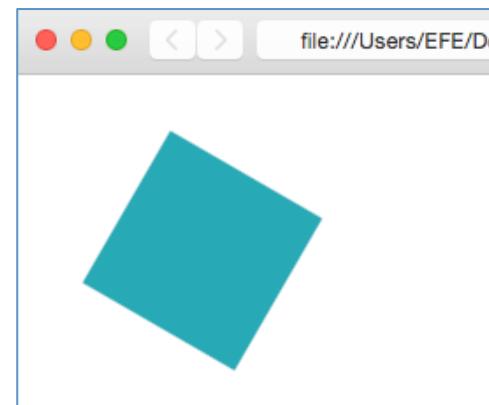
- Distorsion :

```
#bloc{  
    -webkit-transform:skewX(30deg);  
    -moz-transform:skewX(30deg);  
    -o-transform:skewX(30deg);  
    transform:skewX(30deg);  
}
```



- Rotation :

```
#bloc{  
    -webkit-transform:rotate(30deg);  
    -moz-transform:rotate(30deg);  
    -o-transform:rotate(30deg);  
    transform:rotate(30deg);  
}
```



Transformations, transitions et animations

- Les transitions :
 - Les transitions sont de nouvelles propriétés de la norme CSS3 qui permettent de mettre en place des effets (de transition) entre différents états.
 - 4 propriétés :
 - **transition-property** : la propriété qui sera modifiée par la transition
 - **transition-timing-function** : fonction de transition utilisée (ease, linear, ease-in, ease-out, ease-in-out)
 - **transition-duration** : durée de la transition
 - **transition-delay** : temps avant que la transition ne se déclenche

Transformations, transitions et animations

- Les transitions :
 - Exemple :

```
h1{  
    margin : 0 0 50px 30px;  
    color: green;  
    transition-property:color;  
    transition-duration:4s;  
    transition-timing-function:linear;  
}  
  
h1:hover{  
    color:red;  
}
```



Transformations, transitions et animations

- Les animations :
 - CSS3 Animations est une **extension** des CSS3 Transitions
 - Exemple

```
#bloc1{
    width:100px;
    height:100px;
    background-color:green;
    opacity:1;
}

#bloc1:hover{
    animation-name:anim;
    animation-duration:2s;
    animation-iteration-count:infinite;
    -webkit-animation-name:anim;
    -webkit-animation-duration:2s;
    -webkit-animation-iteration-count:infinite;
}
```

```
@keyframes anim{
    0% {opacity:1;}
    50% {opacity:0;}
    100% {opacity:1;}
}

@-webkit-keyframes anim{
    0% {opacity:1;}
    50% {opacity:0;}
    100% {opacity:1;}
}
```

Les Media Queries

- Media queries :

<http://www.w3.org/TR/css3-mediaqueries/>

- Règles à appliquer pour changer le design d'un site en fonction des caractéristiques de l'écran
- Possibilité :
 - De créer un fichier css par règle (ex : "Si la résolution est inférieure à 600px de large, charge le fichier petite_resolution.css"),

```
<link rel="stylesheet" href="normal.css" />
<link rel="stylesheet" media="screen and (max-width: 600px)" href="petite_resolution.css" />
```

- D'écrire directement la règle dans le même fichier css (ex : "Si la résolution est inférieure à 600px de large, charge les propriétés CSS suivantes")

Les Media Queries

- **Media queries :**
 - **Types de média :**
 - **screen** : écran classique
 - **print** : impression
 - **tv** : télévision
 - **projection** : projecteur
 - **all** : tous les types d'écran.
 - **Règles disponibles :**
 - **height** : hauteur (à utiliser également avec min- et max-)
 - **width** : largeur (à utiliser également avec min- et max-)
 - **orientation** : orientation du périphérique (portrait ou paysage).

Les Media Queries

- Media queries :
 - Exemple :

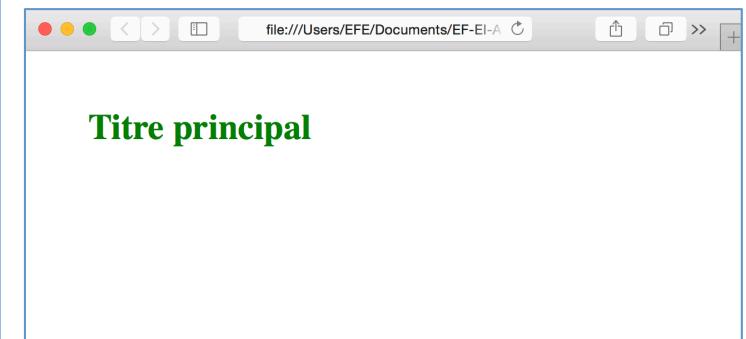
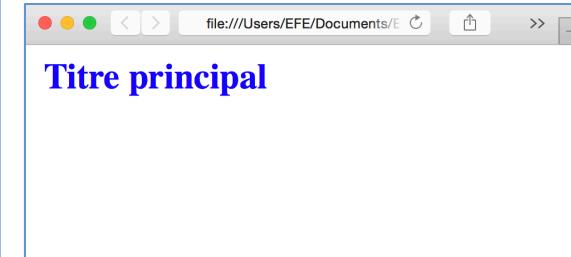
```
<!doctype html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Media Queries</title>

    <!--
    <link rel="stylesheet" href="normal.css" />
    <link rel="stylesheet" media="screen and (max-width: 600px)"
          href="petite_resolution.css" />
    -->

    <style>
        h1 {
            color : green;
            margin: 50px;
        }

        @media screen and (max-width: 600px) {
            h1 {
                color : blue;
                margin: 10px;
            }
        }
    </style>
</head>

<body>
    <h1>Titre principal</h1>
</body>
</html>
```



Les nouvelles API du HTML5

- HTML5 fournit de nombreuses API permettant notamment plus d'interactivité avec l'utilisateur.
- Principales API :
 - Audio et vidéo
 - Géolocalisation
 - Local Storage
 - Canvas
 - Drag and Drop
 - Offline
 - Workers
 - Web Sockets

Les nouvelles API du HTML5

- Les API Audio et Vidéo

http://www.w3schools.com/tags/ref_av_dom.asp

- Même API pour ces éléments "media"

- Méthodes :

- `load()` : charge (ou recharge) l'élément
 - `play()` : joue le media à partir de l'endroit courant (début par défaut)
 - `pause()` : met le media en pause

- Attributs :

- `duration` : durée du média (en secondes)
 - `currentTime` : temps écoulé
 - `muted` : le son est-il activé ou pas (booléen)

Les nouvelles API du HTML5

- Les API Audio et Vidéo
 - Exemple :

```
<article>
  <h2>Fichier mp3 (avec contrôles intégrés)</h2>
  <p>
    <audio id="audio" src="audio/ghostwriter.mp3" controls></audio>
  </p>

  <h2>Contrôles externes (javascript)</h2>
  <p>
    <input type="button" value="Jouer" id="play" />
    <input type="button" value="Pause" id="pause" />
    <input type="button" value="Stop" id="stop" />
    <input type="button" value="Muet" id="mute" />
  </p>
</article>
```

```
$(function(){
  $('#play').click(function(){
    $("#audio").trigger('play');
  })
  $('#pause').click(function(){
    $("#audio").trigger('pause');
  })
  $('#stop').click(function(){
    $("#audio").trigger('pause');
    $("#audio").prop('currentTime', 0);
  })
  $('#mute').click(function(){
    var isMute = $("#audio").prop('muted');
    $("#audio").prop('muted', !isMute);
  })
});
```

Fichier mp3 (avec contrôles intégrés)



Contrôles externes (javascript)

Jouer Pause Stop Muet

Les nouvelles API du HTML5

- Géolocalisation
 - Principalement faite pour les appareils mobiles (meilleure précision)
 - La localisation se fait par l'adresse ip
 - Utilisation de la propriété : `navigator.geolocation`
 - `getCurrentPosition(fonctionCallback)` :
 - Renvoie la position courante
 - `watchPosition(fonctionCallback)` :
 - Notification à chaque changement de position
 - `clearWatch(id)` : stoppe la notification.

Les nouvelles API du HTML5

- Géolocalisation
 - La méthode `getCurrentPosition(...)` renvoie un élément de type position :
 - `coords.latitude`
 - `coords.longitude`
 - `coords.altitude`

```
<body>
  <header>Géolocalisation</header>
  <article>
    <h2>Récupération de la localisation</h2>
    <div class="code">Vérifier les paramètres du navigateur pour autoriser la géolocalisation.</div>
    <br /><br />

    <input type="button" value="Afficher ma position" id="geo" />
    <br /><br />

    <div id="myposition"></div>
    <br />
  </article>
</body>
```

Les nouvelles API du HTML5

- Géolocalisation

```
$('geo').click(function(){
    navigator.geolocation.getCurrentPosition(getPosition);
});

function getPosition(position){
    var lat = position.coords.latitude;
    var lon = position.coords.longitude;
    var alt = position.coords.altitude;

    var infoposition = "Latitude: "+lat+"<br />";
    infoposition += "Longitude: "+lon+"<br />";
    infoposition += "Altitude: "+alt+"<br />";

    $("#myposition").html(infoposition);
}
```

Géolocalisation

Récupération de la localisation

Vérifier les paramètres du navigateur pour autoriser la géolocalisation.

[Afficher ma position](#)

Latitude: 48.8640493
Longitude: 2.3310526
Altitude: null

Les nouvelles API du HTML5

- LocalStorage
 - API de stockage de données
 - Ressemble à la gestion des cookies mais est plus puissante (stockage de 10Mo de données au lieu de 4Ko)
 - Les données sont stockées pour un domaine précis et peuvent récupérées dans n'importe quelle page du domaine en question.

```
window.localStorage.setItem('clé', 'données');  
window.localStorage.getItem('clé');  
window.localStorage.removeItem('clé');
```

Les nouvelles API du HTML5

- Le mode Offline
 - Permet d'indiquer au navigateur qu'il doit stocker des éléments (pages web, css, images, ...) en cache afin qu'ils puissent être affichés offline.
 - Utilisation d'un "manifest" dans lequel on indique les éléments à stocker.
 - Liaison dans la page HTML à ce manifest
 - Indication au serveur (Tomcat, Apache, ...) qu'il doit tenir compte des fichiers manifest

Les nouvelles API du HTML5

- Le mode Offline
 - Le fichier `cache.manifest` :

```
CACHE MANIFEST
# version 1.0.0

CACHE:
# http://www.monserveur.com/monappli/styles.css
monappli/styles.css
monappli/index.html
monappli/scripts.js
monappli/images1.jpg
```

- Le fichier `index.html`

```
<!DOCTYPE html>
<html manifest=".cache.manifest">
<head>
<meta charset="utf-8" />
```

Les nouvelles API du HTML5

- Le mode Offline
 - Le serveur :
 - Ajouter une directive à Apache (ou .htaccess)
AddType text/cache-manifest .manifest
 - Pour Tomcat :
 - Ajouter dans le fichier web.xml :

```
<mime-mapping>
  <extension>manifest</extension>
  <mime-type>text/cache-manifest</mime-type>
</mime-mapping>
```

Les nouvelles API du HTML5

- L'élément Canvas
 - Définit un espace permettant de créer des dessins et des images de manière dynamique.

<canvas> </canvas>

- Récupération du contexte :

```
<article>
  <h2>Canvas</h2>
  <canvas id = "canvas" width="500px" height="500px"></canvas>
</article>
```

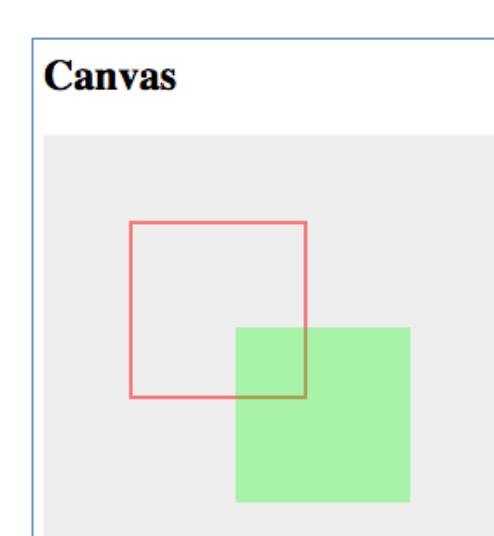
```
var canvas = $('#canvas')[0];
// ou : var canvas = document.getElementById('canvas')
var context = canvas.getContext("2d");
```

Les nouvelles API du HTML5

- L'élément Canvas
 - Fonctionnalités :
 - `strokeRect` : dessine un rectangle vide
 - `fillRect` : dessine un rectangle plein
 - `clearRect` : efface un rectangle

```
context.strokeStyle = "red";
context.strokeRect(50, 50, 100, 100);

context.fillStyle = "rgba(0, 255, 0, 0.3)";
context.fillRect(110, 110, 100, 100);
```

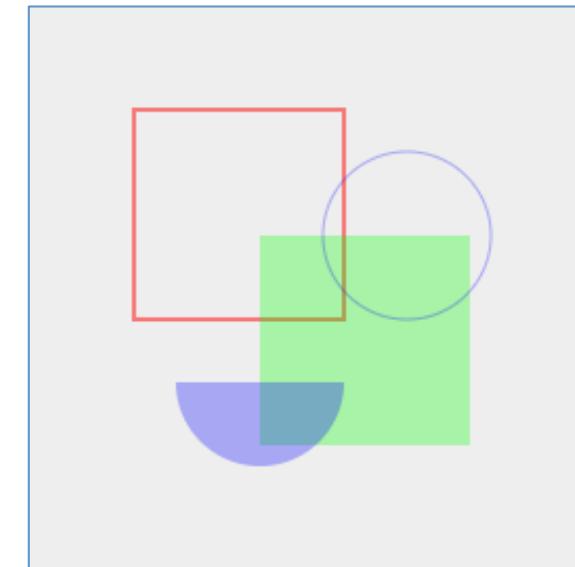


Les nouvelles API du HTML5

- L'élément Canvas
 - Fonctionnalités :
 - `beginPath` : initialise le contexte
 - `closePath` : termine le contexte

```
context.beginPath();
context.strokeStyle = "rgba(0, 0, 255, 0.3)";
context.arc(180, 110, 40, 0, 2* Math.PI);
context.stroke();
context.closePath();

context.beginPath();
context.fillStyle = "rgba(0, 0, 255, 0.3)";
context.arc(110, 180, 40, 0, Math.PI);
context.fill();
context.closePath();
```



Les nouvelles API du HTML5

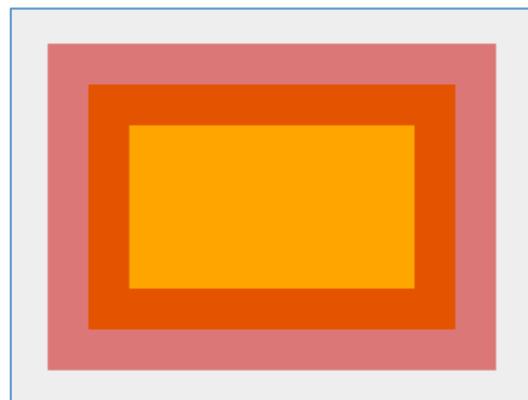
- L'élément Canvas
 - Fonctionnalités :
 - Avec le contexte on peut alors accéder aux propriétés fondamentales
 - `fillStyle` : définit le style de remplissage
 - `lineWidth` : définit l'épaisseur du trait
 - etc.
 - Et aux fonctions fondamentales
 - `fillRect()` : dessine un rectangle plein
 - `lineTo()` : dessine une ligne
 - `drawImage()` : dessine une image
 - etc.

Les nouvelles API du HTML5

- L'élément Canvas
 - Exemple :

```
ctx.fillStyle = 'orange';
ctx.strokeStyle = 'rgba(200, 0, 0, 0.5)';
ctx.fillRect(90, 260, 180, 120);

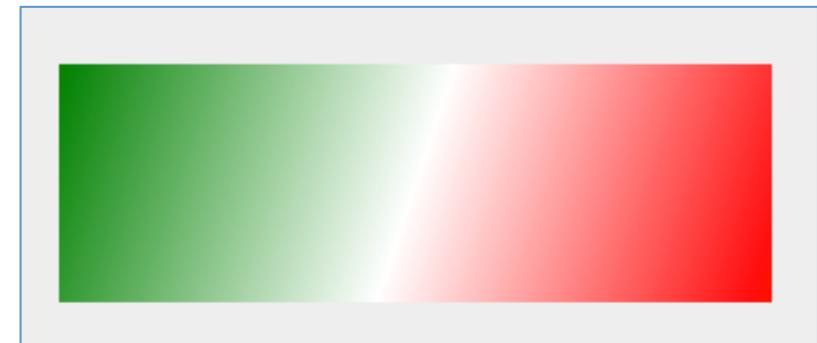
ctx.lineWidth = 40;
ctx.strokeRect(90, 260, 180, 120);
```



Les nouvelles API du HTML5

- L'élément Canvas
 - Le dégradé linéaire :
 - Utilisation de la fonction :
`createLinearGradient(xA, yA, xB, yB)`
avec A point de départ et B point d'arrivée

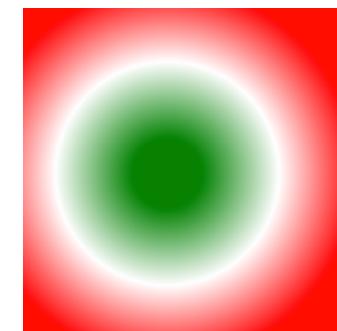
```
var gradient = ctx.createLinearGradient(100, 200, 400, 300);
gradient.addColorStop(0, 'green');
gradient.addColorStop(0.5, 'white');
gradient.addColorStop(1, 'red');
ctx.fillStyle = gradient;
ctx.fillRect(100, 200, 300, 100);
```



Les nouvelles API du HTML5

- L'élément Canvas
 - Le dégradé radial :
 - Utilisation de la fonction :
- createRadialGradient(xA, yA, rA, xB, yB, rB)
- avec A point de départ et B point d'arrivée et r : rayon

```
var gradient= ctx.createRadialGradient (200, 250, 30, 200, 250, 170);
gradient.addColorStop(0, 'green');
gradient.addColorStop(0.5, 'white');
gradient.addColorStop(1, 'red');
ctx.fillStyle = gradient;
ctx.fillRect(70, 100, 300, 300);
```



Les nouvelles API du HTML5

- L'élément Canvas

- Le crayon:

- La forme du crayon au extrémités d'un tracé avec la propriété **lineCap** qui peut prendre les valeurs **butt** (par défaut), **round** ou **square**
 - La forme du crayon aux jointures d'un tracé avec la propriété **lineJoin** qui peut prendre les valeurs **miter** (par défaut), **round** ou **bevel**

```
ctx.beginPath();
ctx.strokeStyle = 'black';
context.lineWidth=15;
ctx.lineCap='round';
ctx.lineJoin="round";
ctx.moveTo(100, 200);
ctx.lineTo(200, 300);
ctx.lineTo(100, 300);
ctx.stroke();
ctx.closePath();
```

