

Développer des composants d'interface

Créer des pages WEB
Feuilles de style CSS

- Introduction
- Différentes versions
- Feuille de style
- Règle CSS
- Définition des styles
- @import : bonnes pratiques
- Notion d'imbrication
- Sélecteurs
- Sélecteurs multiples
- Sélecteurs hiérarchiques
- Sélecteur de classe
- Sélecteur id
- Pseudo-éléments et Pseudo-classes
- Les unités de mesures

- Différentes propriétés
- Propriétés non standards
- Exemples de propriétés : polices et couleurs
- Blocs
- Liens sur internet

Introduction

- La technologie CSS a été définie par le [W3C](#) (World Wide Web Consortium). Cet organisme a pour but de proposer des standards pour le web
- Le W3C est constitué de membres faisant partie de diverses sociétés évoluant dans le monde du Web (Microsoft, Netscape, IBM, Sun, ...). Ces standards sont appelés des recommandations
- Sur le site du W3C vous pouvez consulter en ligne les recommandations CSS. Il existe d'autres recommandations (HTML, HTTP, XML, ...)
- La première recommandation CSS date de décembre 1996

Différentes versions

- CSS 1 1996
- CSS 2 1998
- CSS 2.1 2002 (1^{er} working draft)
- CSS 3 2005 (1^{er} working draft)

Feuille de style

- Quelle que soit la version de CSS, une feuille de style CSS consiste en une liste de déclarations. Elles sont de deux sortes :
 - ⇒ les règles at (@)
 - ⇒ les jeux de règles
- Les règles at commencent par un mot-clé at, composé du caractère "@" immédiatement suivi par un identifiant (ex. '@import', '@page')

- Une règle CSS consiste en deux parties principales :
 - ⇒ Un sélecteur ('h1')
 - ⇒ Une déclaration ('color: blue'). Celle-ci se compose à son tour de deux parties :
 - ⇒ une propriété ('color')
 - CSS2 comporte plus de 100 différentes propriétés
 - ⇒ une valeur ('blue')
- Exemple de règle


```
h1 { color: blue }
```

Définition des styles

- Styles embarqués dans le tag HTML
 - ⇒ Ajouter un paramètre **Style** qui va contenir une chaîne de caractères

```
<p style="color:#808000;font-style:italic;text-align:center">Cool ce style<p>
```

- Styles embarqués dans la page HTML
 - ⇒ Ajouter le couple de tags HTML `<style> . . . </style>` dans l'entête du document HTML

```
<style>
a,h1,h2,h3,p { font-family: Verdana, Arial; font-weight: bold; }
a { color: #0060A0; font-size: 10pt; }
a:hover { color: #000000 }
h1 { color: #004080; text-align: center; }
</style>
```


Définition des styles

- Styles pour un ensemble de pages
⇒ créer un fichier indépendant d'extension ".css" et inclure ce fichier dans les fichiers HTML

Fichier style.css

```
a,h1,h2,h3,p { font-family: Verdana, Arial; font-weight: bold; }  
a { color: #0060A0; font-size: 10pt; }  
a:hover { color: #000000 }  
h1 { color: #004080; text-align: center; }
```

Fichier monfichier.html

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

Définition des styles

- Importer une feuille de style depuis d'autres feuilles de style

```
@import « fichierstyle.css»;
```

ou

```
@import url(« fichierstyle.css»);
```

- Importer une feuille de style par média. Si aucun média n'est spécifié, la feuille de style est importée pour tous les médias (braille, tv, tty, ...)

```
@import url(« impression.css») print;
```

```
@import url(« ecran.css») projection, tv;
```



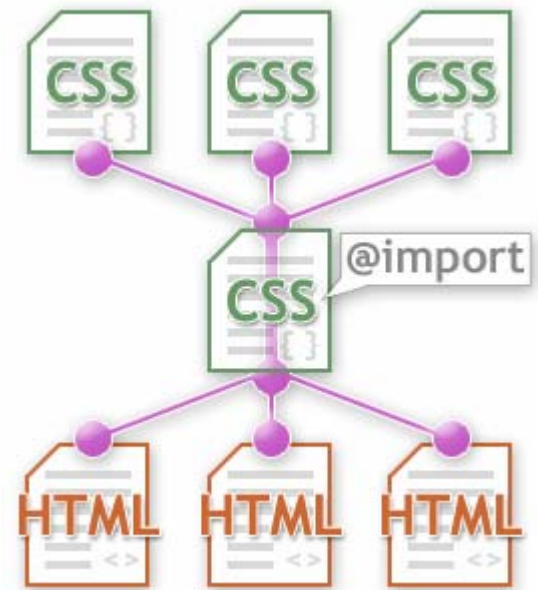
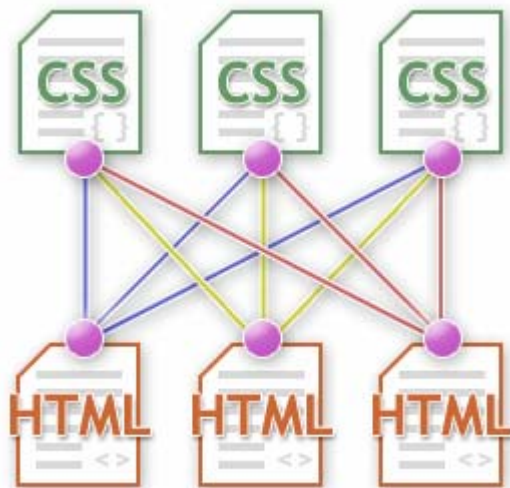
Les anciens navigateurs tel que Netscape 4 ne gèrent pas l'@import

Compatibilité @import / navigateur

Navigateur	Support
Internet Explorer Windows (5.0 – 7.0)	Les règles @import qui spécifient le type de média sont ignorées
Mozilla Firefox (1.5 – 2.X)	OK
Safari (2.0)	OK
Opera (9.0)	OK
Konqueror (3.5)	OK
Internet Explorer Mac	Ne supporte pas la syntaxe @import "style.css" media;

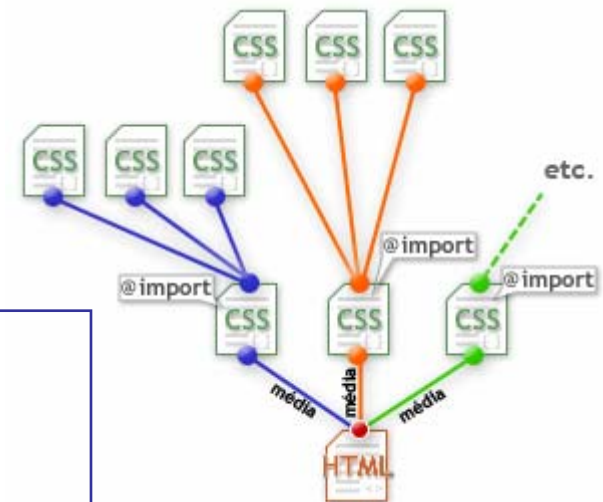
@import : bonnes pratiques

- Lier toutes leurs feuilles de styles via la balise <link />
- Centraliser l'appel aux feuilles de styles dans un seul fichier CSS via @import



@import : bonnes pratiques

- L'utilisation de la règle @import étant limitée pour cause de compatibilités navigateurs, il faut donc :
 - ⇒ Lier des feuilles de CSS via la balise <link /> pour chaque type de média
 - ⇒ Pour chaque média, centraliser vos feuilles de styles en utilisant sans modération la règle @import



```
<link rel="stylesheet" type="text/css" href="fichierstyle1.css" />
<link rel="stylesheet" type="text/css" href="fichierstyle2.css" media="print" />
<link rel="stylesheet" type="text/css" href="fichierstyle3.css" media="tty" />
```

Notion d'imbrication

- Cascading : les caractéristiques de présentations se propagent "en cascade" d'un élément à ses fils
- Si des définitions sont contradictoires, c'est la définition la plus spécifique qui l'emporte

```
<html>
<head>
<style>
p {color:#FF0000}
</style>
</head>
<body>
<p style="color: green">texte1</p>
<p>texte2</p>
<p><font color="blue">texte3</font></p>
</body>
</html>
```

- CSS 1

- ⇒ **Le sélecteur d'id (#)**

Permet de définir une classe unique pour un élément html unique identifié

Éléments concernés : tous

Compatibilité : 

```
#monid { color: red ; border: 1px solid #000000 ; margin-left: 30px ;
width: 200px ; height: 50px ; padding: 5px ; background-color: #ffffff ; }
p#violet { color: rgb(186,18,245) ; text-align: center ; margin-top: 20px
; }
```

```
<div id="monid">Bob Marley</div>
<p id="violet">George Benson</p>
```


- CSS 1

- ⇒ **Le sélecteur de classes (.)**

Permet de sélectionner des éléments d'une même classe définis à l'aide de l'attribut **class**. Ils peuvent être attribués à différents éléments

Éléments concernés : tous

Compatibilité : 

```
.maclasse { background-color: #bcc2fc }
p.violet { background-color: rgb(186,18,245) }
```

```
<div class="maclasse">Bob Marley</div>
<p class="violet">George Benson</p>
```


- CSS 1

- ⇒ **Le sélecteur de type**

Un sélecteur de type permet de sélectionner tous les éléments de son type

Éléments concernés : tous

Compatibilité : 

```
p { font-weight: bold; text-indent: 1em; }
```

```
<p>Le sélecteur de type</p>
```

- CSS 1

⇒ **Le sélecteur descendant**

Un sélecteur descendant est constitué de deux sélecteurs simples séparés par un espace

Éléments concernés : tous

Compatibilité : 

```
div p { font-weight: bold; }
div#section ul li.informations { color: red; }
```

```
<div id="section">
  <p>Le sélecteur descendant</p>
  <ul><li>Element noir</li>
  <li class="informations">Element rouge</li>
  </ul>
</div>
```

- CSS 2

- ⇒ **Le sélecteur universel (*)**

Le sélecteur universel désigne tous les éléments possibles

Éléments concernés : tous

Compatibilité : 

```
div * span { color:red }
```

```
<div>
    <p>janvier<br /><span>février</span></p>
    <span>mars</span>
</div>
```

- CSS 2

- ⇒ **Le sélecteur d'enfants (>)**

Permet de sélectionner un élément à l'intérieur d'un autre, uniquement s'il n'y a pas d'éléments imbriqués intermédiaires

Éléments concernés : tous

Compatibilité : 

```
div > span { color:red }
```

```
<div>
```

```
    <p>janvier</p>
```

```
    <p>février<span> mars</span></p>
```

```
    <span>avril</span>
```

```
</div>
```

- CSS 2

⇒ **Le sélecteur de frères adjacents (+)**

Permet de sélectionner un élément lorsque celui-ci succède directement à un autre

Aucune balise ne doit être présente entre la balise de fermeture du premier élément et celle d'ouverture du second

Éléments concernés : tous

Compatibilité : 🟢

```
div + span { color: red }
```

```
<div>
  <p>janvier <span>février</span></p>
  <span>mars</span>
</div>
<span>avril</span>
```

- CSS 2

- ⇒ **Le sélecteur d'attributs**

Le sélecteur d'attributs permet de sélectionner les balises via leurs attributs. Possibilité de sélectionner des attributs qui contiennent des éléments qui commencent ou terminent par une chaîne déterminée (CSS 3)

Éléments concernés : tous

```
a { color:blue }
a[lang='en'] { color:red }
a[lang='es'] { color:green }
p[attribut^=foo] { color:grey }
p[attribut$=iop] { color:red }
p[attribut*=aze] { color:red }
```

```
<a href="#" lang="fr">un site en français</a>
<a href="#" lang="en">un site en anglais</a>
<a href="#" lang="es">un site en espagnol</a>
<p attribut="foobar">Cette ligne sera grise</p>
```

- CSS 3

- ⇒ **Le sélecteur d'adjacence (~)**

Permet de sélectionner un élément lorsque celui-ci succède directement à un autre. Au contraire du sélecteur +, des balises peuvent être présentes entre la balise de fermeture du premier élément et celle d'ouverture du second

Spécifications sont en cours

Éléments concernés : tous

Compatibilité : 🟡

```
div ~ span { color: red; }
```

```
<div>Les mois</div>
<p>janvier <span>février</span></p>
<span>mars</span></div>
<span>avril</span>
```

Sélecteurs multiples

- Spécifier une règle pour un ensemble de types de tags
- Les tags sont séparés par une virgule
 - ⇒ sélecteur1, sélecteur2 ... { règle1 ; règle2 ; ... }

```
p, h1, h2, h3 { font-family: verdana; }
h1, h2, h3 { font-weight: bold; }
h1 { text-align: center; color: #A0E0FF; }
p { text-align: justify; }
```


Sélecteurs hiérarchiques

- Définir un style pour des tags imbriqués dans d'autres tags : les tags sont séparés via un espace
 - ⇒ **sélecteur1 sélecteur2 ... { règle1 ; règle2 ; ... }**
- Occurrences de sélecteur2 qui sont un descendant d'une occurrence de sélecteur1
 - ⇒ **sélecteur1 > sélecteur2 ... { règle1 ; règle2 ; ... }**

```
table p { par-indent : 0; }
ul > li { margin-left : 10px ; }
```

Sélecteur de classes

- Possibilité d'appliquer un même style sur plusieurs éléments d'un document, même s'ils sont de nature différente
⇒ notion de classe de style
- Une classe de style est définie par un sélecteur qui commence par un point

```
<html><head>
<title>Page de test</title>
<style>
.style1 { color: red; }
</style>
</head><body>
<h1 class="style1">Titre de niveau 1</h1>
<p class="style1">Premier paragraphe</p>
</body></html>
```

Sélecteur id

- Possibilité d'identifier un élément du document HTML
 - ⇒ Ajouter un paramètre **id** au tag considéré : la valeur de ce tag est un nom (identificateur) que vous choisissiez (chaque nom doit être unique dans un document)
 - ⇒ Pour lier le style à cet élément, il faut utiliser un sélecteur dont le nom commence par le caractère **#** suivi de l'identificateur

```
<html><head>
<title>Page de test</title>
<style>
#monId { color: red; }
</style>
</head>
<body>
<p id="monId">Second paragraphe</p>
</body></html>
```

- Quelle peut être l'utilité d'un **id** qui ne sert qu'une fois dans un document alors qu'il existe le paramètre **class** ?
 - ⇒ L'id remplace progressivement "name" en javascript : on appelle dorénavant les objets et calques avec getElementById, ce qui facilite la compatibilité entre tous les navigateurs depuis IE5, NN6, Opera et Mozilla
 - ⇒ L'id peut servir d'ancre nommée
- <body id="top">...aller en haut

Pseudo-sélecteurs

- CSS 1

⇒ **:link**

Permet de définir l'apparence d'un lien lorsque celui-ci n'a pas encore été visité

⇒ **:visited**

Permet de définir l'apparence d'un lien lorsque celui-ci a déjà été visité

⇒ **:hover**

Permet de définir l'apparence d'un élément voulu interactif lorsque celui-ci est survolé

⇒ **:active**

Permet de définir l'apparence d'un élément voulu interactif lorsque celui-ci est actif, en train d'être cliqué

Pseudo-sélecteurs

⇒ **:first-letter**

Permet de créer une lettrine, de définir l'apparence uniquement de la première lettre d'un paragraphe

⇒ **:first-line**

Permet de définir l'apparence uniquement de la première ligne d'un paragraphe

■ CSS 2

⇒ **:after**

Permet d'insérer un élément de contenu automatiquement après l'élément affecté

⇒ **:before**

Permet d'insérer un élément de contenu automatiquement avant l'élément affecté

Internet Explorer ne reconnaît ni **:after** ni **:before**

Pseudo-sélecteurs

⇒ **:first-child**

Permet de cibler le premier élément enfant d'un autre élément, peut s'utiliser avec des classes ou des redéfinitions de balises. La notation se fait sous la forme : `element_parent element_enfant:first-child { }`

⇒ **:focus**

Permet de modifier l'apparence d'un éléments interactif lorsque celui-ci prend le focus.

:focus n'est pas pris en compte par Internet Explorer

⇒ **:lang**

Permet de cibler un élément qui possède l'attribut lang avec une valeur précise

```
a:lang(fr) { color: green; }  
a:lang(en) { color: red; }
```

Les unités de mesures

- Il existe deux types d'unité de longueur

- ⇒ **Les unités de longueurs relatives**

Elles se rapportent à une autre propriété de longueur. Les feuilles de style qui en emploient sont plus facilement adaptable d'un medium à un autre (par exemple d'un moniteur vers une imprimante laser)

- ⇒ **Les unités de longueurs absolues**

in : pouce (inch) est égal à 2.54 cm

cm : centimètre

mm : millimètre

pt : point est égal à 1/72 de pouce

pc : pica est égal à 12 points

Les unités de mesures relatives

- **em** : permet de donner une mesure relative par rapport à la taille de police de l'élément parent. Si par contre la propriété font-size est donnée avec une mention em, celle-ci se rapporte alors à la taille de l'élément parent. Utilisable pour toutes les propriétés acceptant les mention de longueur. Toute valeur décimale doit être indiquée avec un point et non une virgule

```
.texte { font-size: 1.2em; line-height: 1.6em; }
```

- **px** : Le **pixel (picture element)** ou point est l'unité de base d'une image numérique. C'est le point minimal adressable par le contrôleur vidéo

Résolution VGA est de $640 \times 480 = 307\,200$ pixels

Résolution Super-VGA est de $800 \times 600 = 480\,000$ pixels

Résolution XGA est de $1\,024 \times 768 = 786\,432$ pixels

Les unités de mesures

- % : taille relative à la taille de l'élément directement, à celle de l'élément parent ou rattaché. Utilisable pour toutes les propriétés acceptant les pourcentages

```
.texte { font-size: 120%; line-height: 100%; }
```

- **ex** : L'unité 'ex' est définie par rapport à la valeur de la propriété 'x-height'. Cette valeur correspond à la hauteur du caractère "x" minuscule

```
.texte { font-size: 1.2ex; line-height: 1.6ex; }
```

Différentes propriétés

- **Propriétés des polices**
⇒ font, font-family, font-size, font-style, font-variant, font-weight
- **Propriétés de couleur et de fond**
⇒ background, background-attachment, background-color, background-image, background-position, background-repeat, color
- **Propriétés du texte**
⇒ letter-spacing, line-height, text-decoration, text-align, text-indent, text-transform, text-shadow, vertical-align, white-space, word-spacing, quote
- **Propriétés de boîte**
⇒ border, border-color, border-style, border-width, border-top, border-right, border-bottom, ...

Différentes propriétés

- **Effets visuels**
⇒ clip, direction, overflow, visibility
- **Propriétés de positionnement**
⇒ clear, display, float, position, top, ...
- **Propriétés des listes**
⇒ list-style-type, list-style-image, list-style-position, ...
- **Propriétés de génération de contenu**
⇒ content, counter-increment, counter-reset, ...
- **Propriétés d'impression**
⇒ size, marks, page-break-before, page-break-after
- **Propriétés des tables**
⇒ border-collapse, border-spacing, caption-side

Différentes propriétés

- **Propriétés de l'interface utilisateur**
⇒ cursor, outline, outline-color, ...
- **Propriétés auditives**
⇒ azimuth, cue, cue-after, cue-before, elevation, pause, pause-after

Propriétés non standards

- **Les filtres de Internet Explorer**
⇒ filter:alpha(), filter:blur(), filter:chroma(), filter:dropshadow(), filter:fliph(), filter:flipv(), ...
- **Les barres de défilement d'Internet Explorer**
scrollbar-3dlight-color, scrollbar-arrow-color, scrollbar-base-color, scrollbar-darkshadow-color, scrollbar-face-color, scrollbar-highlight-color, scrollbar-shadow-color, scrollbar-track-color

Exemples de propriétés : polices et couleurs

- **font-style** : italic, oblique ou normal
- **font-variant** : small-caps pour les petites capitales, ou normal
- **font-weight** : bold pour le gras, ou normal
- **font-family** : pour choisir la police. Il est possible d'en indiquer plusieurs, séparées par des virgules. Familles génériques : serif, sans-serif, monospace, cursive...
- **font-size** : changer la taille de la police. Valeurs possibles : xx-small, x-small, small, medium, large, x-large, xx-large, ou bien une valeur relative : larger, smaller, ou bien un pourcentage ...
- **color** : pour définir la couleur. Valeurs : #rrggbb (hexadécimal), rgb(r,g,b) (pourcentages ou décimal), nom en anglais.

- Un bloc (un paragraphe, un titre, une table) est entouré de trois zones : le remplissage (padding), une bordure et une marge
- Les unités de longueur disponibles sont : in, cm, mm, pt, px. On peut également utiliser des pourcentages du bloc parent
- **marges** : margin, margin-top, margin-right, margin-bottom et margin-left
- **bordures** : border, border-top, border-right, border-bottom et border-left
- **remplissage** : padding, padding-top, padding-right, padding-bottom et padding-left

Web Devout

<http://www.webdevout.net/>

Cascading Style Sheets (CSS) Snapshot 2007

<http://www.w3.org/TR/2007/WD-css-beijing-20071019/>

CSS Level 1 Specification

<http://www.w3.org/TR/1999/REC-CSS1-19990111>

CSS Level 2 Revision 1 Specification

<http://www.w3.org/TR/CSS21/cover.html#minitoc>

CSS3 Advanced Layout Module

<http://www.w3.org/TR/2005/WD-css3-layout-20051215/>

Cascading Style Sheets Current Work

<http://www.w3.org/Style/CSS/current-work#table>

