

Création de site web

Feuilles de styles

Damien Nouvel

Feuilles de styles

Plan



- Sites, pages, gabarits
- La déportation des styles
- Feuilles de styles en cascade (CSS)
- Principales propriétés CSS
- Les sélecteurs CSS avancés

Feuilles de styles

Plan



- Sites, pages, gabarits
- La déportation des styles
- Feuilles de styles en cascade (CSS)
- Principales propriétés CSS
- Les sélecteurs CSS avancés

Feuilles de styles

Sites, pages, gabarits



- Site : nom de domaine / **ensemble de pages** web
- Page web :
 - Adresse (URL) ?
 - Contenu (HTML) ?
- Site défini par :
 - Adresse (URL) « racine »
 - **Structure du site** : liens entre les pages
 - Fonctionnalités (modules, formulaires)
 - **Charte graphique** :
 - Organisation des éléments graphiques
 - Styles par défaut des éléments (polices, couleurs, etc.)

Feuilles de styles

Sites, pages, gabarits

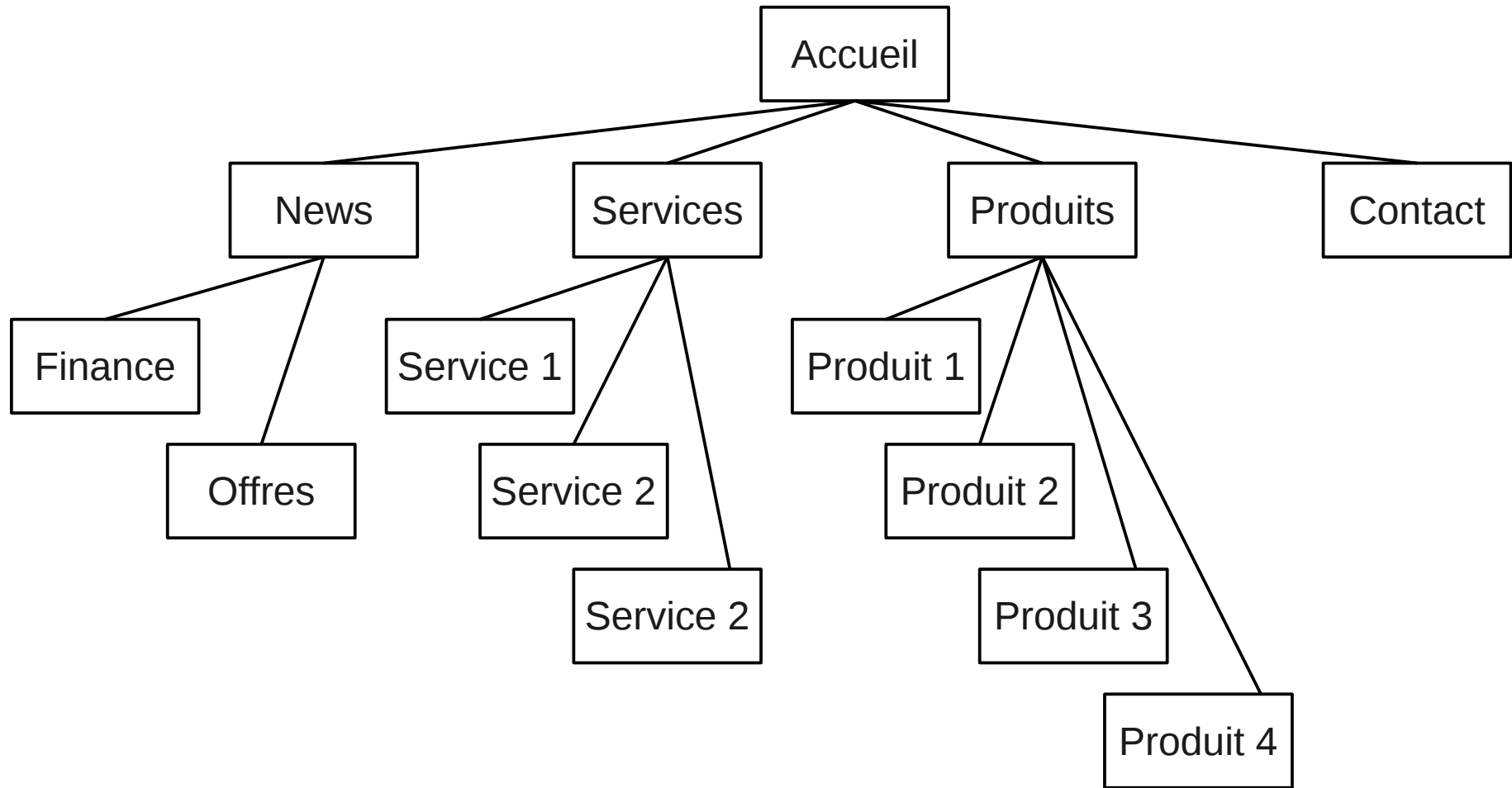


- Structure du site :
 - Généralement hiérarchique (arbre, arborescence)
 - **Dossiers** (URL)
 - Thèmes, catégories, tags, dates, etc.
 - Organisé sous forme de **graphe** :
 - Les **noeuds** : **dossiers**
 - Les **feuilles** : **pages** HTML
 - **Arêtes** entre les noeuds : **liens** principaux
- Le design est généralement le même au sein d'un site :
 - Charte graphique, squelette, gabarit
 - Petites modifications « locales »

Feuilles de styles

Sites, pages, gabarits

- Exemple de structure (site commercial) :

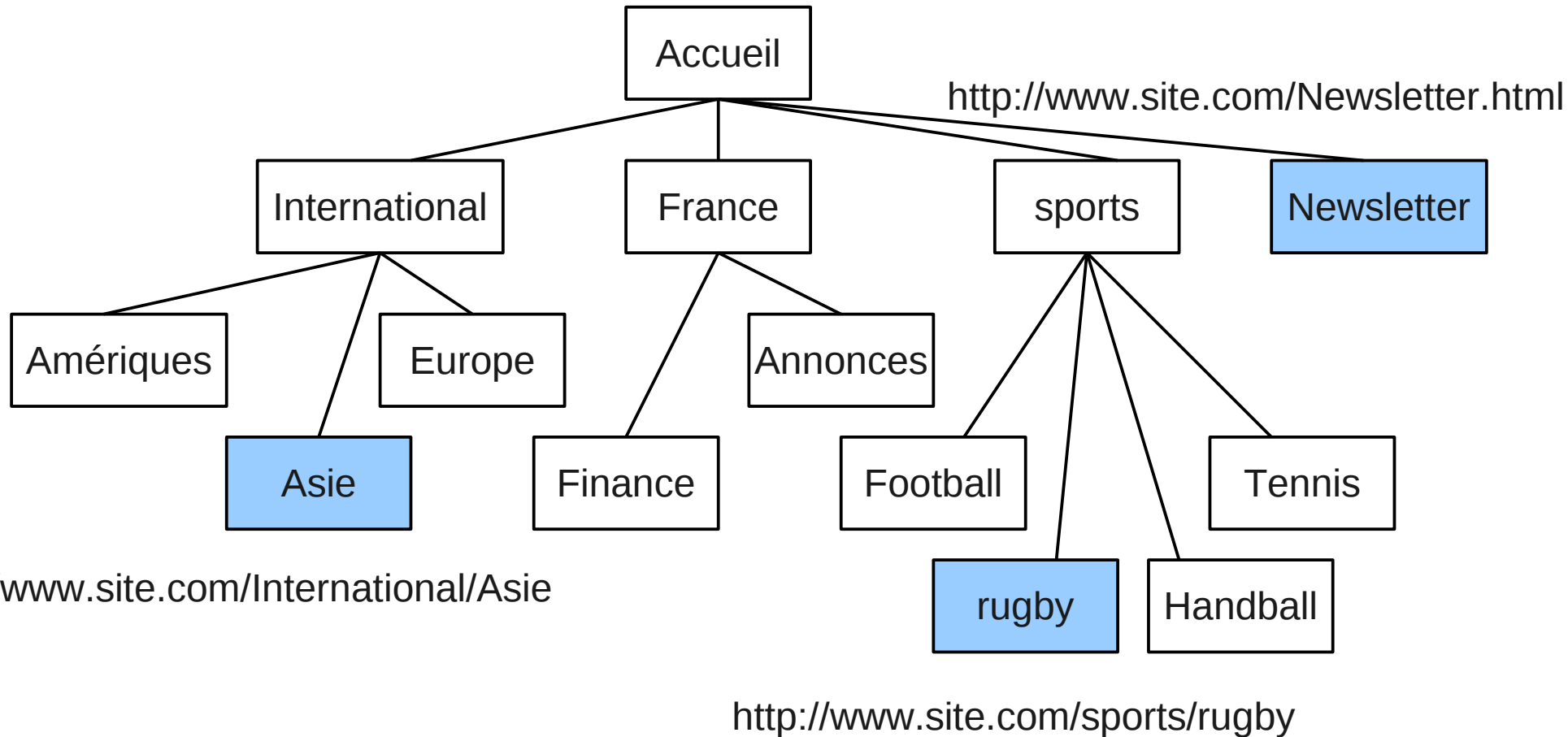


- Adresses de liens (URL) :
 - Utilisation du **slash** «/ » pour **naviguer dans les dossier**
 - **Absolu** :
 - `http://www.domaine.net/dossier1/dossier2/fichier.html`
 - **Relatif** :
 - Même dossier : `"fichier.html"`
 - Sous-dossier : `"dossier/fichier.html"`
 - Dossier « parent » : `"../autredossier/fichier.html"`
 - Fonctionne de même pour tous les liens entre fichiers (HTML, images, CSS, javascript, etc.)
- Fichier par défaut demandé pour un dossier :
 - **"index.html"**

Feuilles de styles

Sites, pages, gabarits

- Exemple d'adresses (site de presse) :

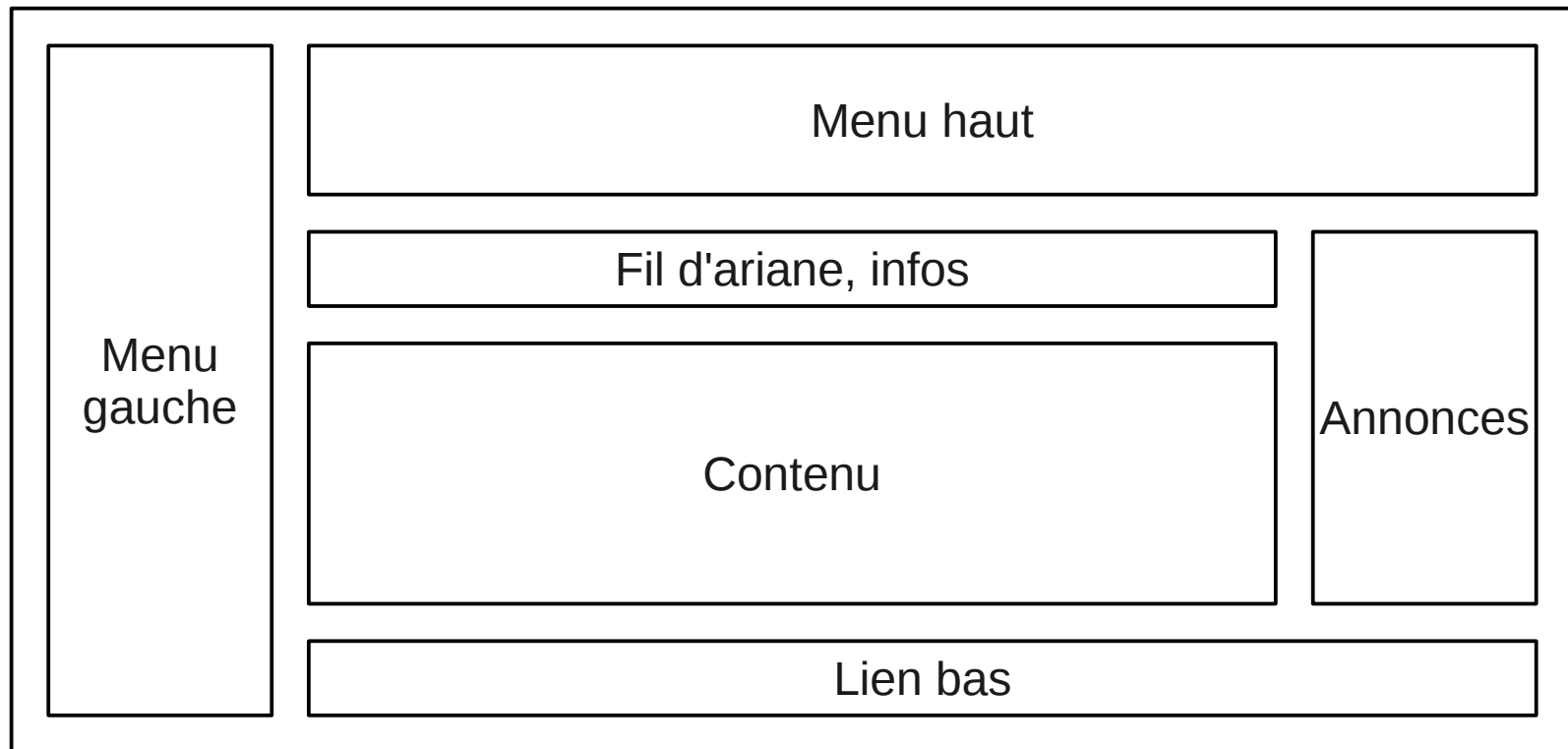


Feuilles de styles

Sites, pages, gabarits

- **Design** (charte graphique, squelette, gabarit) :
 - Généralement prédéfini pour le site, seul le contenu et quelques mises en forme changent

• Par ex. :



Feuilles de styles

Plan



- Sites, pages, gabarits
- La déportation des styles
- Feuilles de styles en cascade (CSS)
- Principales propriétés CSS
- Les sélecteurs CSS avancés

Feuilles de styles

La déportation des styles



- Le XHTML permet de bien décrire la **structure** d'une page web (titres, paragraphes, listes, tableaux, etc.)
- Mise en forme (styles) en XHTML, plus **laborieuse** :
 - Balise « `<font...>...` » à insérer pour **chaque style**
 - Pas vraiment de styles « **par défaut** »
 - Chaque nouvelle page nécessite de **respécifier les styles**
- Des techniques de programmation permettent d'inclure **dynamiquement** des balises de style
 - Inclure un « font » à chaque fois que l'on a un élément HTML particulier (titres, paragraphes, listes, tableaux, etc.)
 - Mais alourdit beaucoup les pages web envoyées par des informations **redondantes**

Feuilles de styles

La déportation des styles



- Le style est souvent une information **commune à tout un site web**, selon :
 - L'**élément HTML** considéré
 - La **place de l'élément**
 - Des éléments **prédéfinis** :
 - Liste rouge, liste bleue, liste à puces, etc.
 - Paragraphe d'introduction, contenu, commentaire, etc.
 - Lien dans le menu, lien dans le texte, lien en bas, etc.
- Intérêt de **centraliser les styles** :
 - Pour une page
 - Pour le site entier (ou une partie du site)

Feuilles de styles

La déportation des styles

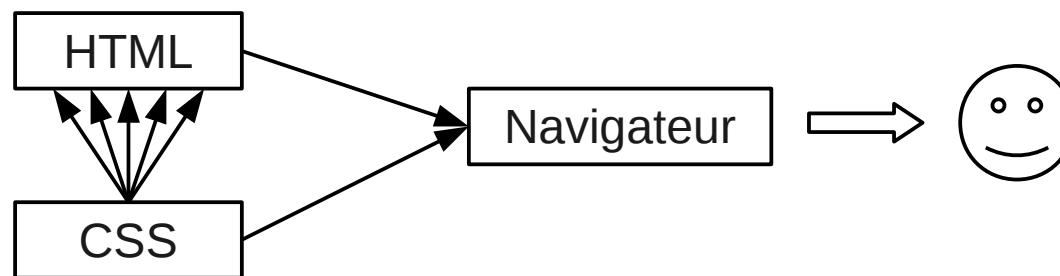


- CSS : Cascading Style Sheets (1995)
 - « Feuilles de styles en cascade »
 - **Feuilles** : centralise les styles
 - **Cascading** : définit les styles les uns par rapport aux autres
- **Trois manières** de spécifier des styles :
 - **Site**, dans un fichier à part, lié en entête (« <head /> »)
 - <link rel="stylesheet" href="monstyle.css" />
 - **Page**, dans le HTML, codé en entête (« <head /> ») :
 - <style type="text/css">...</style>
 - **Élément HTML**, dans n'importe quel élément :
 - <element style="..." />

Feuilles de styles

La déportation des styles

- **Déportation** du style, selon le besoin :
 - Site : **fichier CSS**, utile sur plusieurs pages
 - Page : **entête HTML**, utile uniquement pour la page
 - Élément HTML : **local**, ne sera utile que pour cette balise
- Généralement, il est préférable d'utiliser « **fichier CSS** » (sauf cas particuliers)
- Des règles générales permettent de lier les éléments HTML aux styles définis dans le(s) fichier(s) CSS :



Feuilles de styles

Plan



- Sites, pages, gabarits
- La déportation des styles
- Feuilles de styles en cascade (CSS)
- Principales propriétés CSS
- Les sélecteurs CSS avancés

Feuilles de styles

Feuilles de styles en cascade (CSS)



- Définition des styles pour des éléments **sélectionnés** :
 - Selon le **type d'élément** :
 - `<p />`, `<a />`, `<h1 />`, ``, etc.
 - Selon une **classe** définie :
 - A utiliser dans l'élément HTML avec l'attribut **class** :
 - `<p class="maclassecss">Texte stylé</p>`
 - Plusieurs éléments, peuvent utiliser une même classe de style :
 - `<h1 class="rouge">Titre rouge</h1>`
 - `Lien rouge`
 - Un élément HTML peut avoir plusieurs classes :
 - `<p class="rouge important">Paragraphe important rouge`
 - Selon un identifiant unique au sein de la page (Javascript) :
 - `<p id="monidentifiantcss">Style pour cet élément</p>`

Feuilles de styles

Feuilles de styles en cascade (CSS)



- **Syntaxe CSS :**

- Déclaratif (vs procédural)
- Très différente de HTML !
- Définition de **propriétés** pour des **éléments HTML sélectionnés**, entre accolades « **{** » et « **}** », séparés par « **:** » et « **;** » :

```
selecteur {  
    propriété1 : valeur1 ;  
    propriété2 : valeur2 ;  
    propriété3 : valeur3 ;  
}
```

- Ne tient pas compte des sauts de lignes :

```
selecteur1 { propriété1 : valeur1 ; } selecteur2 { ... }
```

Feuilles de styles

Feuilles de styles en cascade (CSS)



- Sélection CSS :
 - Sélecteurs de base :
 - **Type** d'élément HTML : « p », « h1 », « a », etc.
 - **Classe**, alphanumérique précédé par un « . » : « .classe »
 - **Identifiant**, alphanumérique précédé par un « # » : « #ident »
 - Sélection **conjointe** d'un **type** HTML et d'une **classe** ou d'un **identifiant** :
 - « p.rouge { ... } » → `<p class="rouge">Un paragraphe</p>`
 - « a#lien1 { ... } » → `Mon lien`
 - Sélection **disjointe** d'éléments, séparés par des **virgules** :
 - « a, .important, p.rouge, h3 { ... } » : les liens, éléments de classe « important », paragraphes de classe « rouge », titres de niveau 3 auront tous les propriétés définies

Feuilles de styles

Feuilles de styles en cascade (CSS)



- Principe de la **cascade** :
 - **Héritage** : par défaut, un style appliqué à un élément sera également appliqué aux **éléments qu'il contient (enfants)**
 - **Spécification** : lorsqu'un élément est spécifié par plusieurs styles, les propriétés des styles les plus précis l'emportent :
 - Par ordre de précision : élément HTML < classe < identifiant
- Comportement d'éléments HTML pour le CSS :
 - « **inline** » : texte « **fluide** » (liens, images), seules les propriétés de mise en forme s'appliquent (police, couleurs) et sont généralement **héritées**
 - « **block** » : éléments qui occupent **toute la largeur possible**, qui introduisent un saut de ligne (<div />, <p />, <hX />, , etc.) et qui définissent de surcroît marges et bordures

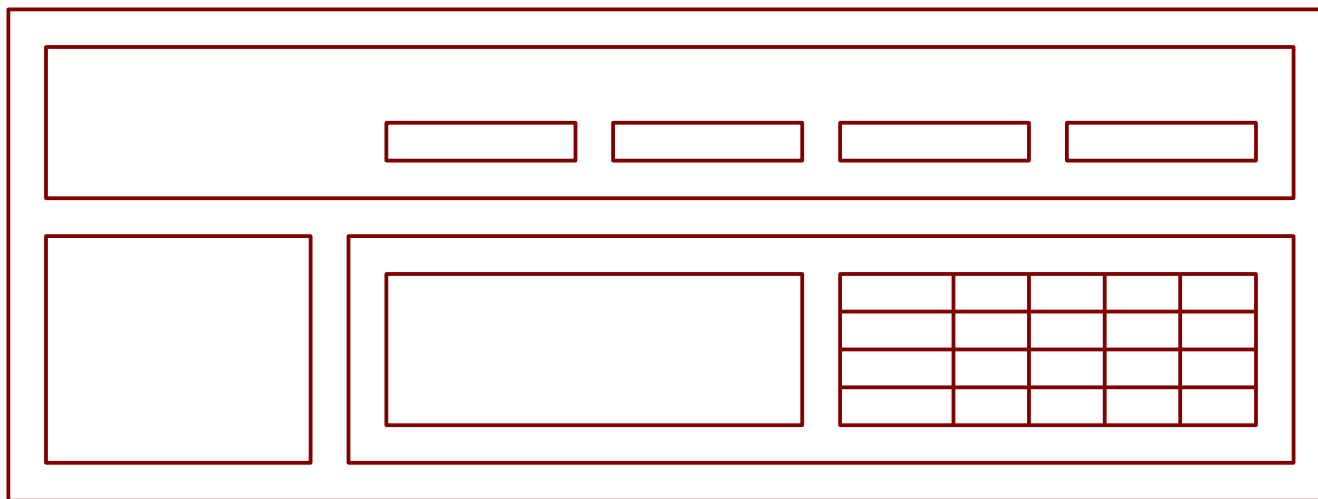
Feuilles de styles

Feuilles de styles en cascade (CSS)

- **Inline** : contenus dans le « flux » de mise en forme :

Il y a dans le flux de **caractères** (ou autres) certains **éléments en ligne** (inline) qui vont être **marqués** d'une **manière particulière et pour lesquels** nous définirons des **propriétés**...

- **Block** : boîtes qui occupent un espace rectangulaire et contiennent d'autres éléments (ou du texte) :



Feuilles de styles

Feuilles de styles en cascade (CSS)



- Modifications apportées par CSS :
 - **Ne plus utiliser **
 - A la place, au besoin utiliser **** avec des styles
 - Quelques éléments à **privilégier** :
 - **** (gras, remplace ****)
 - **** (italique, remplace **<i />**)
 - **<div />** : « division », mise en forme par « **boîtes** »
- Vous ne coderez plus un site de la même manière :
 - **HTML** : structure du site, quelque soit le « look »
 - Références au HTML / CSS qui porte une « **sémantique** »
 - Le design peut changer sans modifier le HTML
 - **CSS** : « **décoration** » d'éléments HTML sélectionnés

Feuilles de styles

Plan



- Sites, pages, gabarits
- La déportation des styles
- Feuilles de styles en cascade (CSS)
- Principales propriétés CSS
- Les sélecteurs CSS avancés

Feuilles de styles

Principales propriétés CSS



- Polices (inline et block, héritées) :
 - **font-family** : nom des polices à utiliser, si disponibles, par ordre de préférence :
 - font-family : Times, Arial, Verdana, serif
 - **font-size** : taille de police :
 - En pixels (px), pourcentage (%), **relatif** (em), point (pt), etc.
 - **font-weight** : « bold », « bolder » ou « lighter »
 - **font-style** : « italic » « oblique »
 - **text-decoration** : « underline », « overline », « strike-through »
 - **font-variant**, **text-transform** : (petites) majuscules
 - **color** : couleur de police, nom (« red ») ou RGB (« #FF0000 »)
 - **background-color** : couleur de fond

Feuilles de styles

Principales propriétés CSS



- Par ex. définir une classe pour un message important :

```
.important {  
    font-family : Arial, sans-serif ;  
    font-size : 2em ;  
    font-weight : bold ;  
    text-decoration : underline ;  
    font-variant : small-caps ;  
    color : #AA0055 ;  
}
```

- Les propriétés « font-XXX » peuvent-être résumées en une seule propriété « font » :

```
font : Arial 2em bold small-caps ;
```


Feuilles de styles

Principales propriétés CSS

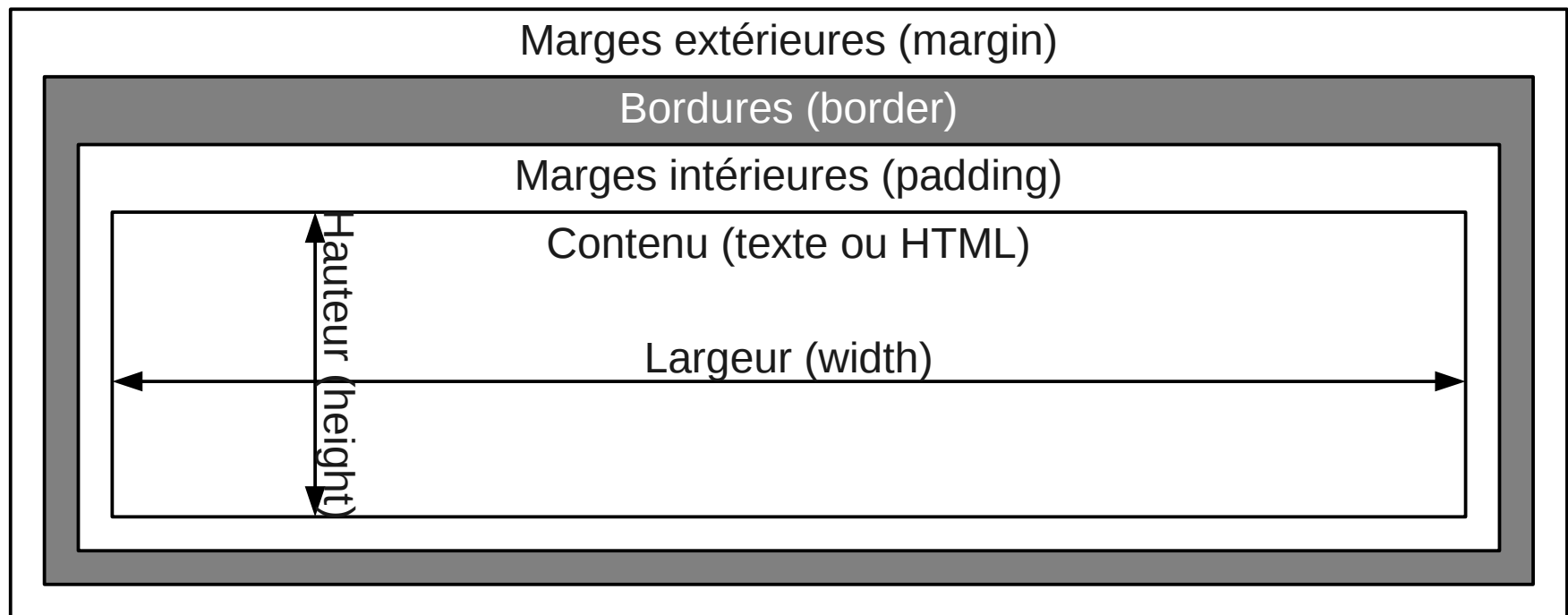


- Les textes sont généralement des éléments « **inline** » définis à l'intérieur d'éléments « **block** » :
 - On peut définir des propriétés du texte **au niveau du block**
 - Définir une **famille de police** au niveau « body » le plus haut
 - Les éléments **inline** spécifient surtout des propriétés de police
 - **Attention** :
 - Les `<p />`, ``, ``, sont (par défaut) des éléments block, tandis que `` est (par défaut) inline
- Alignement du texte **défini au niveau du block** :
 - **text-align** : « left », « right », « center », « justify »
 - **vertical-align** : généralement pour les cellules de tableaux (selon navigateur), « top », « middle » ou « bottom »

Feuilles de styles

Principales propriétés CSS

- Boîtes (**block**, non-héritées) :
 - Tout ce qui est « **rectangulaire** » : `<div />`, `<hX />`, `<table />`, `<td />`, `<p />`, ``, ``
 - Modèle de **boîte** (« box model ») pour les afficher :



Feuilles de styles

Principales propriétés CSS



- Éléments block (suite) :
 - **width** : largeur, en pourcentage (%) ou pixels (px)
 - Par défaut, occupe toute la largeur disponible
 - **height** : hauteur, en pourcentage (%) ou pixels (px)
 - Par défaut, hauteur calculée selon le contenu
 - **border** : style des bordures :
 - Épaisseur (px), style (« solid », « dashed »...) et couleur (RGB)
 - Par ex. : « border : 10px solid #00FFAA ; »
 - Une valeur : vaut pour les quatre bords
 - On peut spécifier :
 - Un bord : border-top, border-right, border-bottom, border-left
 - Une propriété : border-width, border-style, border-color
 - Épaisseur : plusieurs valeurs (haut, droite, bas, gauche)

Feuilles de styles

Principales propriétés CSS



- Éléments block (suite) :
 - **margin** : marges extérieures, en pixels (px) :
 - Distance entre le bord de l'élément et les autres éléments
 - Comme les bords : plusieurs valeurs (haut, droite, bas, gauche)
 - De même, peut-être spécifiée : margin-top, margin-right...
 - **padding** : marges intérieures, en pixels (px) :
 - Distance entre le bord de l'élément et les éléments qu'il contient
 - Comme les bords : haut, droite, bas, gauche
 - De même, peut-être spécifiée : padding-top, padding-right...
 - **background-color** : couleurs (nom ou RGB) :
 - Définit la couleur interne, valable sur le contenu, les marges intérieures, jusqu'au bordures

Feuilles de styles

Principales propriétés CSS

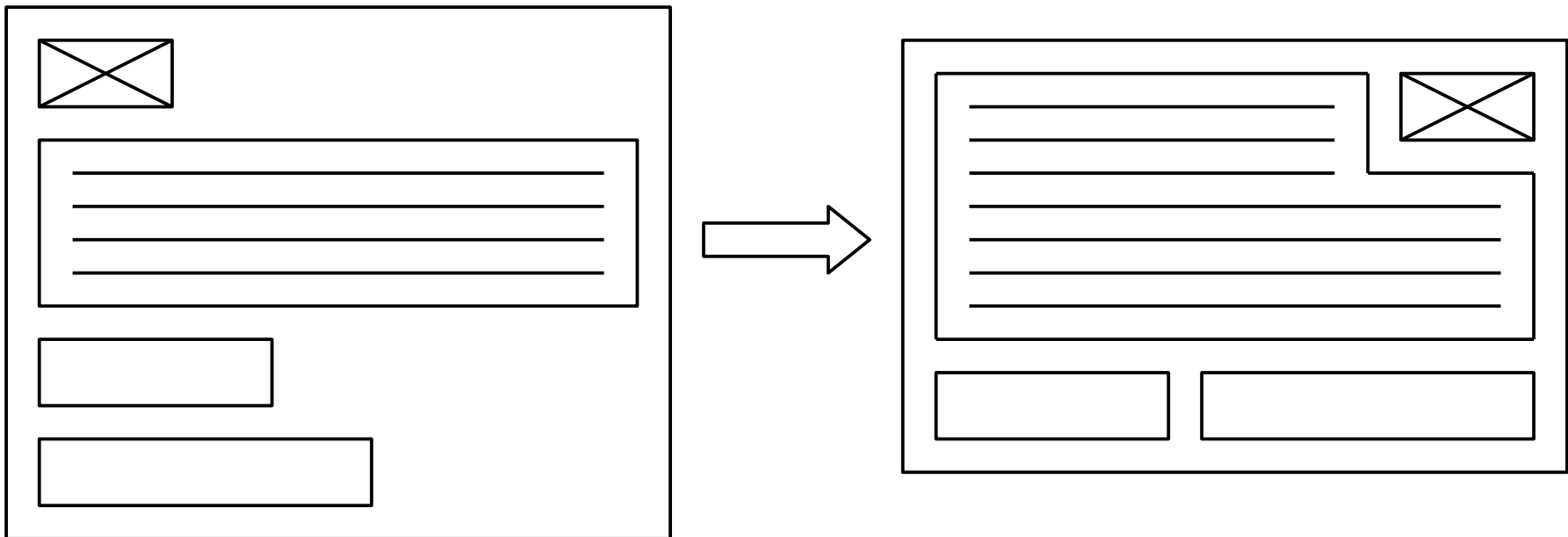


- Éléments block (suite) :
 - Type de positionnement des éléments :
 - **position = relative** : par rapport à sa position calculée
 - **position = absolute** : par rapport au document ou à un parent
 - Coordonnées de positionnement des éléments :
 - **top, bottom, left, right** : en pixels (px) et selon le type de positionnement relatif/absolu
 - Généralement, on **évite d'utiliser ces propriétés** et on préfère **définir correctement les agencements à l'aide de marges et d'éléments flottants**, pour mieux **s'adapter** au poste client :
 - Par ex. : menu en haut, contenu en dessous (marges), menu à gauche (flottant et marges), annonces à droite (flottant et marges), liens en bas (marges)

Feuilles de styles

Principales propriétés CSS

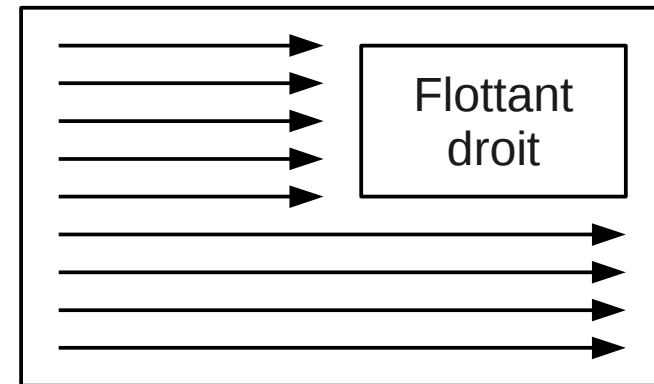
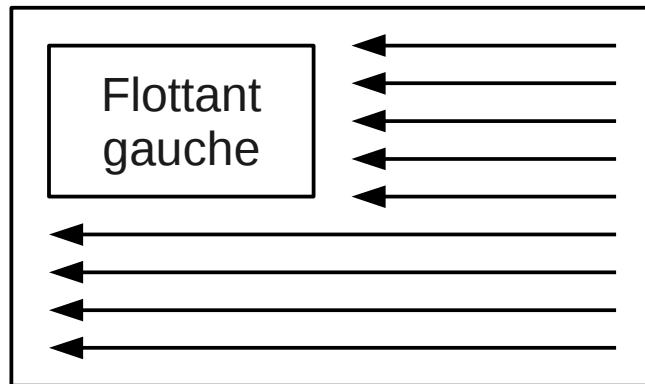
- **Éléments flottants :**
 - Par défaut, les éléments block introduisent un saut de ligne
 - On souhaite des mises en pages dans lesquelles les éléments « flottent » (« coulent) les uns autour des autres :



Feuilles de styles

Principales propriétés CSS

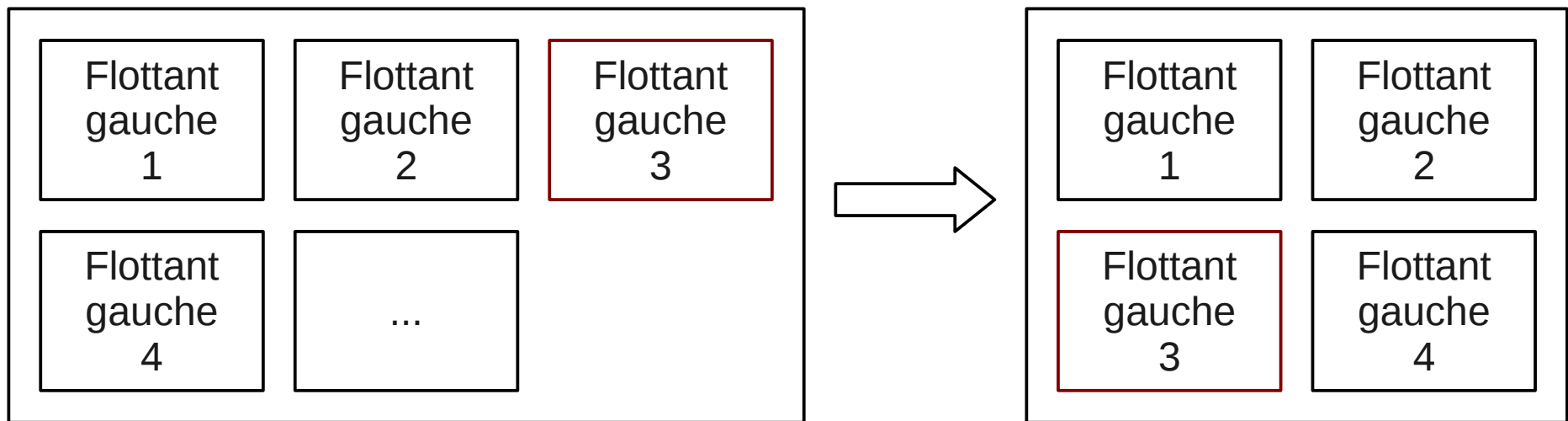
- Éléments **flottants** (suite) :
 - S'applique aux **éléments block**
 - « float : left ; » ou « float : right ; » (placé à gauche ou à droite)
 - Un élément flotte si :
 - Il est mélangé avec des éléments inline
 - Il y a d'autres éléments block flottants
 - A positionner avant l'élément au côté duquel il doit flotter :



Feuilles de styles

Principales propriétés CSS

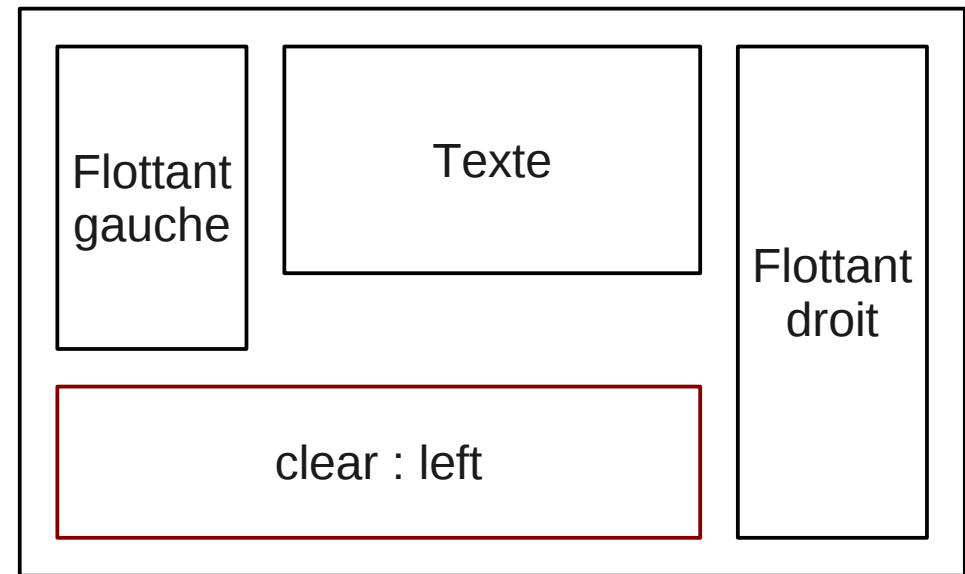
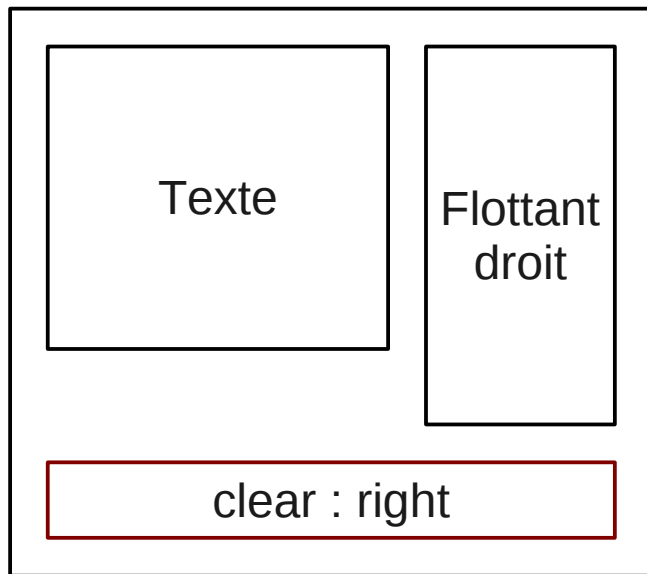
- Éléments **flottants** (suite) :
 - Les éléments flottants **s'adaptent à l'espace disponible** :
 - S'il y a la place, ils l'occupent
 - Sinon ils se placent **en dessous** : permet de disposer des flottants les uns à la suite des autres tout en respectant la largeur de l'écran (anciennement utilisation de tableaux, peu pratiques)



Feuilles de styles

Principales propriétés CSS

- Éléments **flottants** (suite) :
 - Pour placer un élément sous une zone comportant des éléments flottants :
 - « clear: left ; », « clear: right ; », « clear: both ; »,
 - Permet de « passer à la ligne » :



Feuilles de styles

Principales propriétés CSS



- Quelques autres propriétés utiles :
 - Affichage d'éléments :
 - **display** (none, block, inline, list-item) : modifie l'affichage par défaut des éléments
 - **visibility** (hidden, visible) : indique si l'élément s'affiche, mais il occupe en tous cas de la place à l'écran
 - **overflow** (visible, hidden, scroll, auto) : indique que faire lorsque le contenu est plus grand que l'espace disponible pour l'élément
 - **z-index** (entier) : ordre des « calques »
 - Boîtes :
 - **min-width, max-width** (et **-height**) : taille min / max
 - Listes :
 - **list-style-type** (circle, square, decimal, upper-roman) : puces affichées devant les listes

Feuilles de styles

Plan



- Sites, pages, gabarits
- La déportation des styles
- Feuilles de styles en cascade (CSS)
- Principales propriétés CSS
- Les sélecteurs CSS avancés

Feuilles de styles

Les sélecteurs CSS avancés



- Sélecteurs **contextuel** :

- Un sélecteur peut-être composé d'une **suite de sélecteurs** (de base) séparés par des espaces pour appliquer un style à l'élément HTML **le plus imbriqué**, dans le **contexte** représenté par les éléments précédents

- Par ex. :

- « p a { ... } », tous les liens contenus dans un paragraphe :

```
<p> mon paragraphe avec <a>un lien</a> surtout !</p>
```

- « ul span.rouge { ... } », tous les éléments de classe « rouge » contenus dans une liste :

```
<ul>
```

```
<li>Un item <span class="rouge">un peu rouge</span></li>
```

```
<li class="rouge">Item tout rouge</li>
```

```
</ul>
```

```
<p>Ceci n'est<span class="rouge">pas concerné</span>.</p>
```

Feuilles de styles

Les sélecteurs CSS avancés



- Sélecteur **universel** :
 - Tous les éléments (éventuellement dans un contexte) : *
 - $p * \{ \dots \}$: tous les éléments dans un paragraphe
 - Utilisé pour les propriétés qui ne sont **pas héritées**
 - Autant que possible à éviter (effets de bord)
- Sélecteur d'**enfants directs** :
 - Seulement l'élément juste en dessous du contexte : >
 - $p > a \{ \dots \}$: les liens à l'intérieur du paragraphe, mais pas ceux qui sont dans un autre élément au sein du paragraphe
- Sélecteur de **frère adjacent** :
 - Seulement l'élément qui suit : +

Feuilles de styles

Les sélecteurs CSS avancés



- Sélecteurs de **pseudo-classes** :
 - Liens :
 - « **a:visited** » : un lien qui a déjà été visité
 - « **a:hover** » : lorsque la souris « glisse » sur le lien
 - « **a:link** », « **a:active** », « **a:focus** »
 - Premiers enfants :
 - « **:first-child** » : premier élément enfant
 - p a:first-child : premier lien à l'intérieur d'un paragraphe
- De nombreux autres, dont ceux définis en CSS2 (et diversement interprétés par les navigateurs) :
 - attributs « **img[title]** », « **img[title=Mon image]** »
 - « **:lang** », « **:first-line** », « **:first-letter** », « **:before** », « **:after** »