



# Introduction aux feuilles de style avec le langage CSS

Dr. Dellys Hahchemi

Cours Technologie et Développement Web (TDW) 2CS

2019-2020



2

# Introduction

# Qu'est-ce que le CSS ?

- ▶ Un langage de règles permettant d'affecter un style à des éléments
- ▶ Ce style peut être une couleur, une police de caractères, une position, ...etc.
- ▶ Les règles peuvent être déclarées à l'intérieur du fichier HTML ou à l'extérieur en utilisant un fichier CSS séparé

# Syntaxe

Sélecteur

Déclaration

p {

border : 1px solid red;  
font-family : Arial;

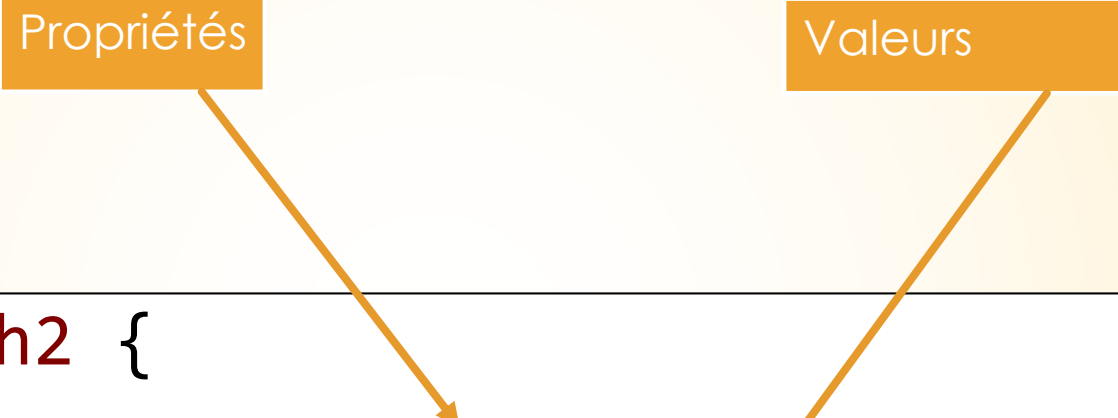
}

# Syntaxe, suite

Propriétés

Valeurs

```
h1, h2 {  
    margin: 5px;  
    font-family :Arial;  
    color :Yellow;  
}
```



# CSS intégré dans le fichier HTML

- Le CSS peut être intégré dans le HTML en utilisant la balise « style »
- La balise a un attribut « type » qui dit être à « text/css »

```
<style type="text/css">
  p {
    font-family: 'Trebuchet MS';
    margin-bottom: 5px;
  }

  h1, h2 {
    color: blue;
  }
</style>
```

## Fichier CSS externe-La balise link

- Un fichier HTML peut référencer plusieurs fichiers CSS
- Il vaut mieux mettre les feuilles de style dans des fichiers externes pour faciliter la maintenance

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

# Sélecteurs CSS



# Introduction

- ▶ Les sélecteurs permettent de sélectionner des éléments ayant un critère particulier : par exemple tous les éléments, appartenant à la même balise, même classe,...
- ▶ Les sélecteurs CSS sont sensibles à la casse

# Sélecteurs CSS

Sélecteur	Description	Exemple
Sélecteur universel	Sélectionne tous les éléments. Utilise le symbole « * »	* {} : tous les éléments
Sélecteur de type	Concerne tous les éléments appartenant à la même balise	<b>h2</b> {} : tous les « h2 » <b>p, span</b> {} tous les paragraphes et les spans
Sélecteurs de classe	Concerne tous les éléments appartenant à une certaine classe (identifiés par l'attribut « classe »). Un élément HTML peut appartenir à plusieurs classes.	<b>span.nuance</b> : tous les span dont la classe est « nuance » <b>.couleurVerte</b> {} : n'importe quel élément ayant la classe « couleurVerte »
Sélecteur d'id	Concerne un seul élément identifié par son attribut « id »	span#spanId {} : le span qui a l'attribut id à « spanId » #unElement {} : n'importe quel élément ayant comme id « unElement »

# Sélecteurs CSS - Suite

Sélecteur	Description	Exemple
Descendant direct	Les éléments qui sont descendants directs d'un autre élément. Le symbole « > » est utilisé. La fonction nth-child spécifie quel élément choisir.	<b>div -&gt; p</b> : paragraphes descendants directs d'une div <b>div-&gt;a:nth-child(2)</b> : le deuxième lien descendant direct d'un div <b>div-&gt;a:nth-child(2n+1)</b> : tous les 2nème + 1 liens descendants directs : 1 <sup>er</sup> , 3 <sup>ème</sup> , 5 <sup>ème</sup> ,...etc
Descendants indirects	Tous les éléments qui sont des enfants (pas forcément directs) d'autres éléments	<b>div span.spanInterne</b> : les span avec la classe « spanInterne » qui sont des descendants d'un div
Frère direct	Un éléments adjacent à un autre élément et qui sont au même niveau (symbole +)	<b>h2+p</b> : le paragraphe qui vient directement après un h2
Frère indirect	Les éléments se trouvant au même niveau qu'un autre élément (symbole ~)	<b>li~li</b> : un élément adjacent à un autre élément de liste. Exclut le premier.

# Sélecteurs avancés

Sélecteur	Description	Exemple
:link	Liens pas encore visités	
:visited	Lien visité	
:hover	Élément qui est survolé par la souris	
:focus	Élément qui détient le focus	<b>input:focus</b> : contrôle qui reçoit le focus
[attribut=valeur]	Sélectionne des élément ayant des attributs avec une certaine valeur	<b>a[target=_blank]</b> {} : sélectionne les liens qui ouvrent dans une nouvelle fenêtre
[attribut^=valeur]	Sélectionne les éléments ayant un attribut commençant par une valeur	<b>a[href^=http]</b> : sélectionne les liens ayant une URL commençant par HTTP
[attribut*=valeur]	Sélectionne les éléments ayant un attribut contenant la valeur	<b>a[href*=ynt]</b> : sélectionne des liens qui contiennent « ynt » dans l'URL
:only-child	Sélectionne l'élément qui est l'unique descendant d'un autre élément	div p:only-child : sélectionne le paragraphe unique à l'intérieur d'un div

# Sélecteurs avancés

Sélecteur	Description	Exemple
<code>nth-child(n)</code>	Nème descendant	<code>n-child(2)</code> : deuxième descendant <code>n-child(2n+1)</code> , tous les descendants impairs
<code>:first-child</code>	Premier enfant	
<code>:first-line</code>	Première ligne d'un paragraphe	
<code>:first-letter</code>	Première lettre d'un paragraphe	
<code>:before</code>	Ajoute du contenu avant l'élément	
<code>:after</code>	Ajoute du contenu après l'élément	
<code>not</code>	Sélectionne les éléments qui ne sont pas un élément donné	<code>p &gt; not(a)</code> tous les éléments sauf les liens

# Cascade

- Si plusieurs règles s'appliquent un élément, les mécanismes suivants sont appliqués
  - Si deux règles sont identiques, la dernière est prise
  - Si une règle est plus spécifique que l'autre, elle est prise sur l'autre. `h1` est spécifique sur `*`. `span#id` est spécifique à `span`. `span.maClasse` est spécifique à `.maClasse`.
  - Si on veut prioriser une règle, on la marque avec le mot clé « !important »

# Exercice

1- Donner la signification des sélecteurs suivants:

- `Div p {};`
- `h2#byName, span.byClass {};`
- `div#byName > img:nth-child(2n) {};`
- `p#byName > b: hover {};`
- `div#byName > p:nth-child(2n+1) span[class*=re] {};`
- `article p+p.description:nth-child(2n+1) {};`
- `p > html+head -> span:nth-child(5n+7) img[src*=f] div: hover {};`

# Solution

- ▶ **Div p {};**

Tous les paragraphes qui descendent d'un « div » (c'est un espace et non ne virgule et les paragraphes ne sont pas nécessairement des descendants directes (d'où l'importance de l'arbre pour un début))

- ▶ **h2#byName , span.byClass {}**

Le header identifié par « byName » et les « span » de classe « byClass »

- ▶ **div#byName>img:nth-child(2n) {};**

Les images qui sont des descendants directs dans une position paire des éléments « div » identifiés par « byName »

- ▶ **p#byName>b:hover {};**

Les éléments « b » survolés, qui sont un descendant directe des paragraphes identifiés par « byName »



# Solution

➤ **`div#byName>p:nth-child(2n+1) span[class*="re"] {};`**

Les éléments « Span » qui appartiennent à des classes contenant la séquence "re" et qui descendent des paragraphes de position impaire qui sont des descendants directs d'éléments « Div » identifiés par « byName ».

➤ **`article p+p.description:nth-child(2n+1) {};`**

Les paragraphes de position impaire appartenant à la classe « description » qui sont des frères directs d'un paragraphe qui est un descendant d'un article.

➤ **`p>html+head -> span:nth-child(5n+7) img[src*=f] div:hover {};`**

Le sélecteur est sémantiquement incorrecte pour un document HTML bien formé.

# Exercice

2-Classer ces sélecteurs selon leur priorité:

1. Div p { }
2. Div p > img.maClass { }
3. Div P > Img#ID
4. Div P[color=red] >Img.maclass
5. Div P[color=red] >Img#ID

Solution:

5-3-4-2-1

# Affichage des éléments

- La propriété « visibility » permet d'afficher ou de cacher un élément. Elle peut avoir les valeurs « visible » ou « hidden ».
- La valeur « collapse » de « visibility » s'applique aux tableaux et aux colonnes et permet de cacher une ligne ou une colonne sans affecter la disposition du tableau.
- Un élément peut aussi être caché en utilisant la propriété « display » avec la valeur « none ». La différence avec visibility est que la valeur « hidden » laisse l'espace de l'élément réservé tandis que la valeur « none » le cache et ne réserve pas sans espace.

# Positionnement relatif

- Une position relative déplace un élément par rapport à sa position « normale »
- Pour indiquer une position relative, utiliser la règle « position » avec la valeur « relative »
- Pour bouger l'élément par rapport à sa position normale, utiliser « top », « left », « right » ou « bottom ».
- Le déplacement relatif a parfois par conséquence, de mettre un élément sur un autre (le premier couvre le deuxième). Pour soigner l'effet, utiliser la propriété « z-index ».

# Positionnement

21

<b>Positionnement absolu</b>	<p>Lorsqu'une position absolue est attribuée à un élément, il est positionné par rapport aux coordonnées (top, left, right et bottom) et l'élément n'a plus aucun impact sur les autres.</p> <p>Lorsqu'un élément est positionné d'une manière absolue à l'intérieur d'un autre élément en mode relatif, il est positionné par rapport à l'élément parent.</p>
<b>Positionnement fixe</b>	<p>La position fixe est semblable au positionnement absolu sauf que l'élément garde toujours la même position lorsque l'utilisateur défile la page</p>
<b>Éléments flottants</b>	<p>Un élément flottant est pris de sa position « normale » puis mis à gauche ou à droite.</p> <p>Les autres éléments s'adaptent pour contourner l'élément flottant</p>

# Les transitions et les transformations (CSS3)

# Les transitions

- ▶ Les transitions CSS3 permettent aux développeurs d'effectuer des animations simples sans passer par JavaScript
- ▶ Le principe des transitions est de changer la valeur d'une propriété CSS d'une manière douce et progressive ce qui donne l'effet d'une animation
- ▶ Exemples : changer de couleur, de dimension, ...etc.



# Propriétés

Propriété	Description
transition-property	Propriétés à animer. Peut être « <b>all</b> » pour toutes les propriétés animables ou « <b>none</b> » pour aucune. Peut aussi contenir plusieurs noms de propriétés à condition que ces propriétés soient animables.
transition-duration	Durée de la transition. Spécifiée en secondes ( <b>s</b> ) ou millisecondes ( <b>ms</b> ).
transition-timing-function	Fonction d'interpolation, gère la vitesse de l'animation.
transition-delay	Retard ou avance la transition (en secondes ou millisecondes)
transition	Un alias pour les paramètres de transition



# transition-timing-function

Propriété	Description
ease	Rapide au début, lent à la fin
linear	Vitesse constante
ease-in	Lent au début, rapide à la fin
ease-in-out	Lent au début et à la fin
steps	Exécute l'animation en plusieurs étapes discrètes. Par exemple steps (5,start) ou steps (3,end) début ou fin.

# Propriétés animables

- ▶ La liste des propriétés animables est disponibles sur <http://www.w3.org/TR/css3-transitions/#properties-from-CSS->
- ▶ Pour qu'une propriété soit animable, le navigateur doit être capable de faire des interpolations dessus. Par exemple les couleurs et les dimensions sont animables.

# Transformations

- ▶ Les transformations sont des actions géométriques qui sont effectués sur des éléments HTML
- ▶ Il existe deux types de transformations : en deux dimensions et en 3 dimensions
- ▶ Les transformations 2D sont plus largement supportées que les transformations 3d

# Transformations

Propriété	Description
none	Pas de transformation
translate(x,y)	Translation deux dimensions
translate3d(x,y,z)	Translation 3d
translateX(x), translateY(y) et / ou translateZ(z)	Translation sur l'un des trois axes
scale(x,y)	Mise à l'échelle. Une valeur 1 correspond à la même taille, 0.5 à la moitié et 2 double
scale3d(x,y,z)	Mise à l'échelle 3 dimensions
scaleX(x), scaleY(y) ou scaleZ(z)	Mise à l'échelle sur l'un des trois axes

# Transformations - Suite

Propriété	Description
<code>rotate(a)</code>	Où a est un angle, rotation d'un élément
<code>rotate3d(x,y,z,a)</code>	Rotation 3 dimensions
<code>rotateX(x)</code> , <code>rotateY(y)</code> ou <code>rotateZ(z)</code>	Rotation sur l'un des trois angles
<code>skew(a,b)</code>	Où a et b sont des angles. Inclinaison bidimensionnelle de l'élément
<code>skewX(x)</code> ou <code>skewY(y)</code>	Inclinaison sur l'un des deux axes
<code>matrix(n1..n6)</code>	Transformation sur matrice 2d
<code>matrix3d(n1..n16)</code>	Transformation matrice 3d
<code>perspective(n)</code>	Perspective 3d

# Origine des transformations

- ▶ La propriété transform-origin définit l'origine de la transformation
- ▶ Syntaxe : transform-origin : axe-x axe-y axe-z
- ▶ Par défaut, les valeurs sont 50% 50% 0

# Animations (CSS3)

# Introduction

- ▶ Contrairement aux transitions qui animent une seule propriété sur une période, les animations sont constituées de plusieurs cadres et chaque cadre anime une ou plusieurs propriétés
- ▶ Une animation contient au moins deux cadres
- ▶ Les navigateurs utilisent le GPU pour les animations



# Déclaration des cadres

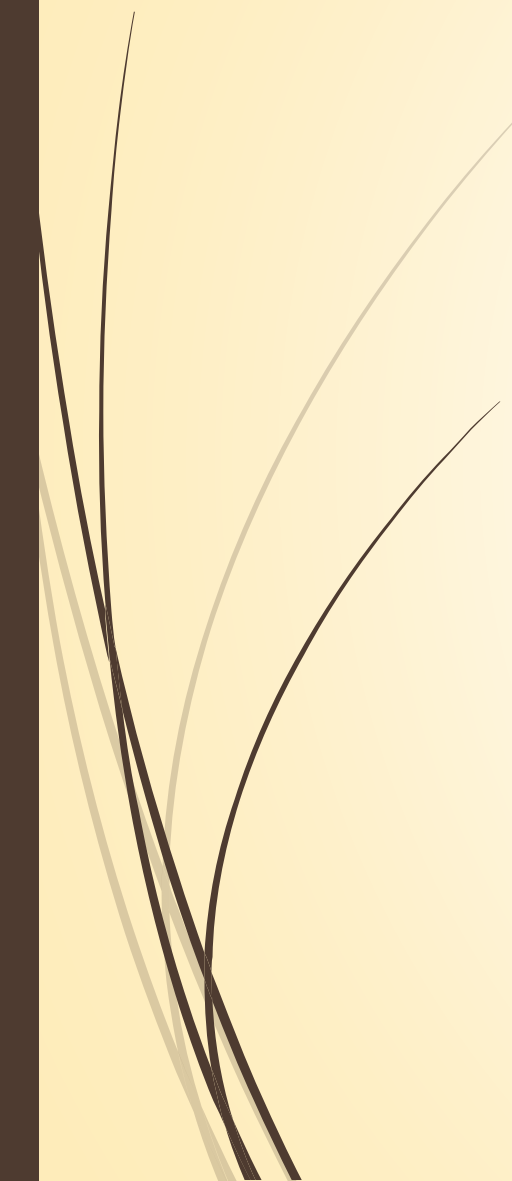
```
@keyframes nom_de_l_animation  
{ from  
    { propriété_à_animer: valeur_initiale; }  
  50%  
    { propriété_à_animer: valeur_intermédiaire; }  
  to  
    { propriété_à_animer: valeur_finale; }  
}
```

# Propriétés d'animation

Propriété	Description
animation-name	Nom de l'animation
animation-duration	Durée de l'animation
animation-iteration-count	Nombre d'itérations. « infinite » signifie une animation infinie.
animation-timing-function	Fonction d'interpolation utilisée pour l'animation. Les mêmes que celles des transitions.
animation-direction	Direction de l'animation : normal, alternate, reverse ou alternate-reverse
animation-play-state	Statut de l'animation (running, paused)
animation-delay	Délai de lecture

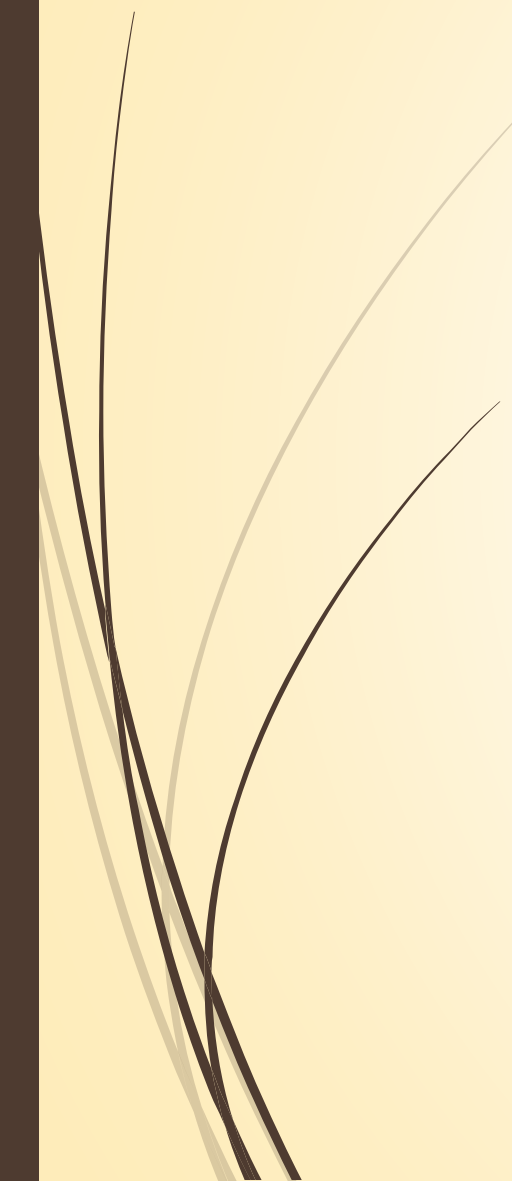


## Exercice 1

- Créez un fichier CSS pour le site de l'entreprise.
  - Créez un sélecteur pour les liens qui descendent d'une liste lors des survoles. La couleur de ces liens devient vert.
  - Créez un sélecteur pour la première ligne et un pour la première colonne d'un tableau. La couleur de fond de ces ligne doit être grise claire.
  - Le titre de la page doit être grand, en gras, en italique et utilisant une police fantaisie.
- 

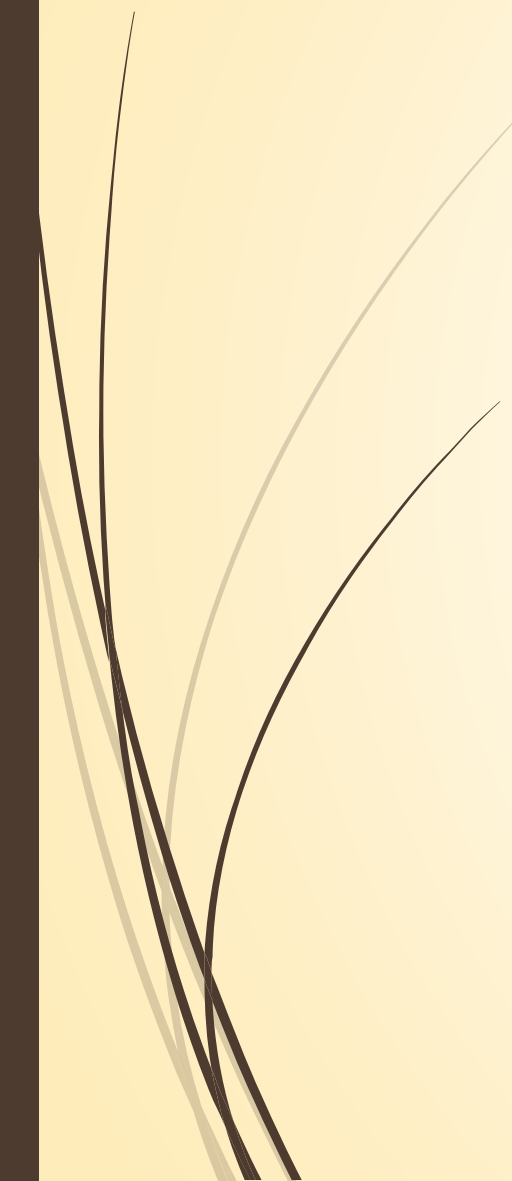


## Exercice 2

- Mettez des liens présentons les offrent de formations en dessous de l'image d'accueil de manière horizontale pour formé un menu, le menu doit apparaitre tous le temps dans la fenêtre active.
- 



## Exercice 3

- Remplacez l'image d'accueil par un diaporama de 4 images.
  - Chaque image prend 50% de la largeur de la page.
  - Les images sont séparés par un vide de couleur noir de 20px.
  - Le diaporama défile de gauche à droite avec une vitesse relativement lente.
  - Le diaporama tourne en boucle et seule le survol de celui-ci permet de l'arrêter.
- 

# Bibliographie

- ▶ Jon Duckett, HTML & CSS, Wiley Editions
- ▶ Tutoriaux Alsacr  ation
- ▶ « Introduction aux animations CSS3 », David Rousset