

Développement et Architecture Web

CSS

R.Gosswiller

- 1 Le CSS
- 2 Syntaxe du CSS et blocs
- 3 Propriétés principales du CSS
- 4 Responsive design

Le CSS

En quelques mots

Definition (wikipédia)

Les feuilles de style en cascade, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML.

- Language côté client
- Basé sur les balises HTML
- Language descriptif et peu organisé

Intégration CSS

Ajouter du CSS à une page HTML

- Directement dans le html head : `<style></style>`
- Importé dans le html head : `<style>@import url("style.css");</style>`
- Fichier séparé (mais renseigné dans le html head) :
`<link rel="stylesheet" type="text/css" href="style.css" />`

Le debugger du navigateur

Utilisé pour vérifier et valider du code html et css

- Structure HTML
- CSS stylesheets
- Console d'erreurs JavaScript

Se lance avec F8 ou F12 selon les navigateurs

Syntaxe du CSS et blocs

Exemple

```
1 body {  
2     background-color: #FB3224;  
3 }
```


Syntaxe

Syntaxe

- La syntaxe est faite pour fonctionner en blocks
- Chaque block à un nom suivi de { }
- Les propriétés CSS sont définies entre les { }
- Chaque propriété est terminée par ;

Exemples de propriétés : background-color, font-weight, font-family...

Travail en blocs

Blocs en HTML

- Un bloc se construit avec `<div>` `</div>`
- La balise définit un rectangle dans la page
- Propriétés de balise : `id`, `class`..
- Attributs de balise : `width`, `height`..

Id et Class

id

Une balise `<div>` peut posséder un id qui est alors unique

Un id est référencé en CSS à l'aide de `#`

class

Une class peut être affectée à plusieurs balises `<div>`

Une class est référencée en CSS à l'aide du symbole `.`

Id et Class

HTML

```
1 <div id="test"> MyText</div>
2
3 <div class="reddiv">Text1</div>
4 <div class="reddiv">Text2</div>
```

CSS

```
1 #test {color : red;}
2
3 .reddiv{color: red;}
```

Cumul des classes

```
1 <div class="c1 c2 c3 c4"></div>
```

Cumul des précisions

Le fonctionnement en blocs permet d'avoir une hiérarchie de balises. On peut alors appliquer du CSS aux balises spécifiques d'un id ou d'une classe spécifique.

Cumul des classes

```
1 .class td {}  
2 #id.class span {}
```

Selectors

Selectors

```
1 [class^="toto"] {} /* Begins with */
2
3 [class$="test"] {} /* Ends with */
4
5 [class*="te"] {} /* Contains */
6
7 input[type="text"] {}
```

Propriétés principales du CSS

Unités

Tailles

- % : width, margins, ..
- pixels(px) : small margins, gaps
- pouces(em), points(pt) : fonts, radiuses

Couleurs

- RGB 0-255 : rgb(0, 0, 0)
- Hexadecimal : #00FF32 (red 00, green FF, blue 32)

Background

Propriétés

- background-color : color
- background-image : url("path/to/img") ;
- background-position : left/right/center top/center/bottom ;
- background-repeat : repeat/repeat-x/repeat-y/no-repeat/initial/inherit ;
- background : combination ;

Texte

Propriétés

- color : color ;
- font-weight : normal/bold/italic/... ;
- font-size : x pt ;
- font-family : verdana, Arial, "Times New Roman" ;

Bordures

Propriétés

- `border-size : x px ;`
- `border-color : color ;`
- `border-style : none/hidden/dotted/dashed/solid/... ;`
- `border-radius : x em ;`
- `border(-top/left/right/bottom) : combination ;`

Alignement

Propriétés

- `padding(-top/left/right/bottom) : x px ;`
- `margin(-top/left/right/bottom) : x px ;`
- `width : x px/% ;`
- `height : x px/% ;`
- `float : left/right/top/bottom ;`

Cadres (frames)

Propriétés

- `box-shadow` : `x px y px z px color` ;
- `opacity` : `0.00 - 1.00` ;
- `position` : `absolute/relative/fixed/initial/...` ;
- `display` : `block/inline/flex/...` ;
- `visibility` : `hidden/visible/collapse` ;

Couleurs

Propriétés

- Keywords : blue, white, yellow, ... ;
- Hexadecimal : #RRGGBB ;
- Hexadecimal simplified : #RGB ;
- RGB : rgb(0-1, 0-1, 0-1) ; or rgb(0-255, 0-255, 0-255) ;
- RGBA : rgba(0-1, 0-1, 0-1, 0-1) ;
- Others : HSL, HSLA, NSColor, ... ;

Gradients

Propriétés

- `linear-gradient(color, color) ;`
- `linear-gradient(position, color, color) ;`
- `linear-gradient(angle, color, color) ;`
- `repeating-linear-gradient(color, color) ;`

Egalement

- `-webkit-` : Webkit-based browsers (Chrome, Safari)
- `-moz-` : Firefox and mozilla-based
- `-o-` : Opera

Responsive design

Le Responsive Design

Problème

Comment adapter une page web à différents navigateurs ?

- Ordinateur de bureau
- Smartphone
- Tablettes

Solution

Les media-queries

Media-Queries

Définition

Les media-queries sont des éléments qui déclenchent des modifications du CSS en fonction de l'environnement client.

Utilisation

Modifier la taille d'un <div> en fonction de l'écran

Changer la couleur d'un texte pour un affichage plasma

S'adapter à un rétro-projecteur

...

Exemple

```
1 @media screen and (max -width: 640px)
```

Media-Queries

Syntaxe

Declaration d'une media-query : @media

Declaration du type : screen, all, tv, ...

Declaration des contraintes : max-width, min-width, and, ...

Types de media-queries

Types

- screen : computer screens
- print : designed for printed communication
- handheld : smartphones and small devices
- aural : vocal synthesis
- braille : blind people
- embossed : printed braille
- projection : To be projected on slides, cinemas, ...
- tty : console
- tv : television
- all : all the elements

Exemple

Exemple

```
1 #test {  
2     width :95%;  
3 }  
4  
5 @media screen and (max -width: 640px) {  
6     #test {  
7         width :80%;  
8     }  
9 }
```

Bootstrap

Framework

Bootstrap est un framework CSS + Javascript conçu pour automatiquement générer des pages HTML/CSS en Responsive Design. Il fonctionne avec des listes de classes appliquées à des blocs.

Origine

Développé en 2011

Conçu pour twitter

Conclusion

Conclusion

- Le CSS applique des propriétés graphiques à des balises HTML
- Pour les appliquer au bon endroit, on utilise une hiérarchie de classes, id, et balises
- Il existe de (très) nombreuses propriétés
- Comme le HTML, il s'agit d'un langage interprété !