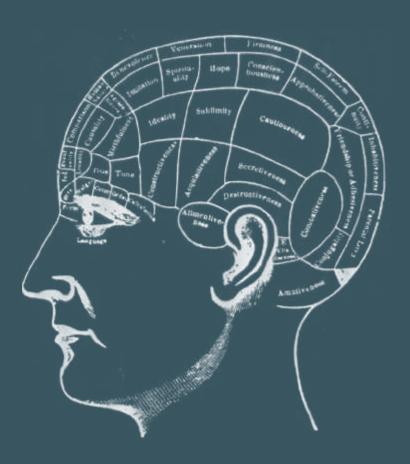


### Méthodologie pour le Design CSS

### La philosophie



## Avec la source (X)HTML, la CSS fait partie des nos livrables

Nos livrables sont le miroir de notre compétence

### La feuille de styles est un document vivant

- Plusieurs « éditeurs »
- Durée de vie considérable
- Les propriétés et leurs valeurs ne sont pas importantes ...
- ... Les sélecteurs, eux, le sont

Réf.: Méthodologie pour l'écriture du code (X)HTML, points 5.1 et 5.2

#### Le design CSS doit être flexible et évolutif

- Tester avec des contenus « extrêmes »
- Zoomer
- Considérer les traductions

Réf.: Méthodologie pour l'écriture du code (X)HTML, point 4.6

## Si le design CSS est la « déco », le (X)HTML sont les fondations d'une maison

Le design CSS ne vit sans le code source, mais le code source vit sans le design CSS

Réf.: Méthodologie pour l'écriture du code (X)HTML, point 2.1

### Toujours aller du plus générique au plus spécifique

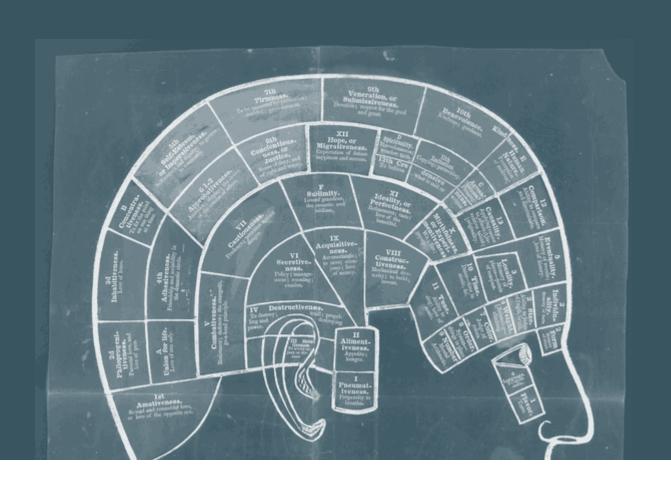
- Avoir toujours le plus d'éléments de design possible avant de commencer
- Eviter de commencer la production par la page d'accueil ou par les blocs spécifiques (navigation, par exemple)
- Réfléchir en termes de composants

Réf.: Méthodologie pour l'écriture du code (X)HTML, points 4.7 et 4.8

### Le cadrage du projet est primordial

- Comportement de la page
- Besoins en termes d'accessibilité
- Exigences client
- Particularités du design
- Versions
- CMS
- CSS d'impression

#### Les Conventions



### Renseigner correctement la feuille de styles

- Nom du projet, client, rôle de la CSS, auteur, date
- La relation avec d'autres CSS
- Index des groupes de styles
- Index des codes couleurs

```
Worldline PlavLive / Atos Worldline
             Main styles definitions
           By Le Studio, Atos Worldline, 2009
/* NOTES: Used in combination with "ie6hacks.css" and "ie7hacks.css"
style-sheets, called by conditional comments by each HTML file --
check those CSS for corrections for IE6 and IE7 browsers
Check out the "dev.css" style-sheet also for modifications or
add-ons done by the dev team
*/
/* MAIN COLORS REFERENCE:
   main color/red:
                     #COO:
   light grey: #CCC;
   dark grey:
                     #666;
#/
/* CSS index (!):
 DEFAULT VALUES
 PAGE BODY STRUCTURE & MAIN ELEMENTS
 NAVIGATION
 TABLE SPECIFICS
 FORMS SPECIFICS
  SEARCH
 GENERAL BLOCKS
 SPECIFIC PAGES
 EXCEPTIONS & SPECIFICS (used sitewide)
```

## Considérer l'ordre et la structure

- Grouper les styles de même nature, et commenter chaque groupe
- Utilisez les espaces et la tabulation pour faciliter la consultation et l'édition
- Garder un même ordre de déclaration des propriétés à chaque définition de style

```
------ PAGE BODY STRUCTURE & MAIN ELEMENTS ------- ! PAGE BODY STRUCTURE & MAIN ELEMENTS
/* --- MAIN BLOCKS --- */
/* Main container */
#container {
    width: 980px;
   margin: 0 auto;
/* Columns wrapper */
#wrapper {
   float: left:
   width: 980px;
   margin-bottom: 20px;
/* Columns */
#mainCt {
   float: left;
   width: 940px;
   min-height: 300px;
   color: #333;
   background: #E3E3E3;
    padding: 20px;
#aside {
    float: left:
   width: 956px;
   min-height: 300px;
   color: #333;
   background: #FFF url(../imgs/bg aside.png) bottom left no-repeat;
    padding: 20px 12px;
   border-top: 3px solid #333;
```

#### Un groupement rationnel

Redéfinition des valeurs par défaut

Structure de page et principaux éléments

Navigation

Styles spécifiques aux tableaux

Styles spécifiques aux formulaires

Les blocs généraux

Styles spécifiques à une page

Autres (popups, modales, éléments interactifs, etc.)

Exceptions & styles spécifiques utilisés sur tout le site

#### Le nommage de styles

Microformats?

Langue?

Comment assurer la sémantique?

Comment considérer les nouvelles balises HTML5?

Réf.: Méthodologie pour l'écriture du code (X)HTML, point 4.4



#### Les Impératifs

## Privilégier les déclarations raccourcies

```
Par exemple :
margin: top/bottom right/left;
color: #FFF;
```

# Toujours définir des tailles de polices relatives et...

- Définir une taille générique pour le BODY de 101%
- Définir les tailles en *em*
- Ajuster si nécessaire en utilisant des pourcentage

#### ... les définir pour les éléments bloc

- Le contrôle sur les tailles est plus précis si elles sont définies au niveau des blocs « primaires » (les plus proches du texte):
- P, DD, DT, LI, TH, TD, CAPTION, etc.
- Et pas : DIV, TABLE, FORM, DL, UL, etc.

#### Lister les pseudo sélecteurs pour la balise a dans le bon ordre

**LoVeHA**te

a

a:link

a:visited

a:hover

a:active

#### Pas de guillemets pour les URLs

background: #575757 url(../imgs/bg\_body.png) repeat-x;

# Toujours donner une largeur à un bloc positionné « absolute »



# Toujours donner une largeur explicite à un bloc « float »

- « auto » est une valeur acceptable
- Les images ont une largeur implicite



### Ne pas donner d'unité à la valeur 0

C'est tout simplement inutile

## Toujours définir une famille de polices générique

Arial, Helvetica, Sans-serif

### Considérer les limites de support d'IE6 pour les sélecteurs

- Pas de classe1.classe2 {...}
- Pas de input[type=submit] {...}



### Les Bonnes Pratiques



## Utiliser un navigateur fiable\* pour le design CSS

...Tester en suite avec tous les autres navigateurs

#### Commencer par égaliser

Egaliser les valeurs par défaut des propriétés des balises :

```
body, div, dl, dt,
  dd, ul, ol, li,
  h1, h2, h3, h4, h5, h6,
  pre, code, form, fieldset,
  legend, input, textarea,
  p, blockquote, th, td {
    padding: 0; margin: 0;
} /* reset */
```

# Réfléchir avant de grouper des sélecteurs dans une même définition

- Gain en Ko, mais perte de flexibilité
- A éviter si les sélecteurs sont utilisés dans de contextes différents
- Une bonne analyse des composants avant le design CSS peut être beaucoup plus utile pour gagner en poids et en structure

Réf.: Méthodologie pour l'écriture du code (X)HTML, points 4.7 et 4.8

### Ne pas laisser de définitions ou de propriétés inutiles

- Les propriétés de test doivent être supprimées
- Vérifier les propriétés qui « ne font rien » ou qui sont redondantes

```
strong {font-weight: bold;}
```

### Utiliser des largeurs en pourcentage toujours que possible

Flexibilité et évolutivité :

on change la largeur du père et les enfants le suivent

### Cacher correctement les liens d'aide à la navigation

- display: none; ne sera pas lu par certains synthétiseurs vocaux
- Utiliser plutôt une marge supérieur/gauche négative

Réf.: Méthodologie pour l'écriture du code (X)HTML, point 5.11

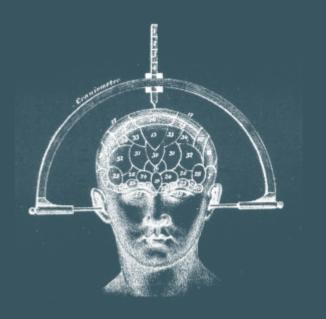
# Attention avant d'aligner verticalement un élément

- line-height doit être de la même auteur que l'élément-père
- Définir l'hauteur plutôt en *em* ou *ex* pour ne pas bloquer la possibilité de zoomer

## Garder une même logique pour définir les espacements

Définissez toujours les marges inférieures et droites pour tous les blocs (en suivant la logique d'assemblage de page)

# Pour le design « liquide », toujours déterminer une largeur max et mini



#### Le DEBOGAGE

Non, vous ne pouvez pas vous en passer de FireBug!

#### Que des bons reflèxes...

- Le DOCTYPE est là, est au bon endroit et est le bon
- Je considère les bugs d'IE
- J'ajoute une couleur de fond pour mieux visualiser le problème

## Validate the F\*%!?\$\page!

### « Hey, kid! T'as pensé au bug des doubles marges? »



# Pour IE6 et IE7, considérez la propriété HASLAYOUT

