

e-application - 2020

1. Les bases du web & de PHP

Matthieu Kowalczyk

Ingénieur en technologies de l'information @CGI

Nathan Levy

Ingénieur en technologies de l'information @CGI



matthieu.kowalczyk@cgi.com
nathan.levy@cgi.com

CGI



UNIVERSITÉ
DE MONTPELLIER

Présentation

- Présentation de l'intervenant.
- Présentation du module.
 - 4 séances.
 - 1 devoir à rendre à la fin.
 - 1 devoir à faire (essentiellement) durant les séances (TP / TD).

Agenda

1. HTML/CSS/JS

1. HTML
2. CSS
3. La performance du Web et les bonnes pratiques YUI
4. Introduction à jQuery
5. Sémantique Web et SEO
6. TP n°1

2. PHP

7. Présentation du langage PHP
8. Installation & configuration
9. Syntaxe PHP
10. TD n°1
11. TP n°2
12. POO
13. Normes et PSR
14. Les outils
15. TP n°3

1 - HTML5

- HTML : passé, présent & futur
- Les normes
- Les balises HTML
- La structure d'un document HTML
- Les avancées d'HTML5

L'HTML5 c'est quoi ?

HTML5 (***HyperText Markup Language 5***) est la dernière révision majeure du HTML (format de données conçu pour représenter les pages web). Cette version a été finalisée le 28 octobre 2014. HTML5 spécifie deux syntaxes d'un modèle abstrait défini en termes de DOM : **HTML5** et **XHTML5**. Le langage comprend également une couche application avec de nombreuses API, ainsi qu'un algorithme afin de pouvoir traiter les documents à la syntaxe non conforme. Le travail a été repris par le W3C en mars 2007 après avoir été lancé par le WHATWG. Les deux organisations travaillent en parallèle sur le même document afin de maintenir une version unique de la technologie. Le W3C clôt les ajouts de fonctionnalités le 22 mai 2011, annonçant une finalisation de la spécification en 2014, et encourage les développeurs Web à utiliser HTML 5 dès ce moment.

Dans le langage courant, **HTML5** désigne souvent un ensemble de technologies Web (HTML5, CSS3 et JavaScript) permettant notamment le développement d'applications (cf. DHTML). ©Wikipédia

HTML, un langage de balises

```
<p>Ceci est un paragraphe.</ p>  
<p>Ceci est un paragraphe avec du <b>gras</ b>.</ p>  
<p id="subtitle">Ceci est un paragraphe avec un ID.</ p>  
<p class="title">Ceci est un paragraphe avec une classe.</ p>  

```

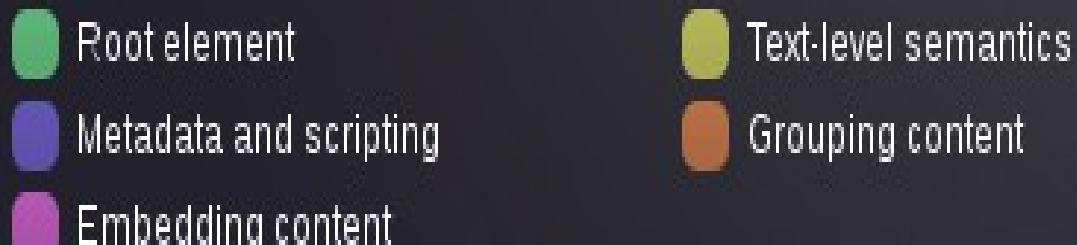
Notez qu'en HTML5, on peut prendre certain raccourcis avec la syntaxe mais je suis de la vieille école ;-) et j'utilise une syntaxe dite xHTML.

Structure d'une page HTML5

```
<!doctype html>
<html>
  <head>
    <title>Titre du document</title>
    <!-- Autres balises de métadonnées -->
  </head>
  <body>
    <!-- Corps du document -->
  </body>
</html>
```

Liste des balises HTML5

- Il existe pléthore de sites ou de livres référençant et décrivant les différentes balises HTML (~107). Rodolphe Rimelé y consacre près de 600 pages dans sa bible sur l'HTML5 !
- [La représentation de Josh Duck](#) était intéressante (archive en PJ) car elle catégorisait et se basait sur une représentation connue de tous basée sur les travaux de [Mendeleiev](#) en chimie.
- Elle est remplacée par [MDN](#).
- Catégorisation des balises :



Periodic Table of the Elements

html																col	table				
head	span											div	fieldset	form	body	h1	section	colgroup	tr		
title	a											pre	meter	select	aside	h2	header	caption	td		
meta	rt	dfn	em	i	small	ins	s	br	p	blockquote	legend	optgroup	address	h3	nav	menu	th				
base	rp	abbr	time	b	strong	del	kbd	hr	ol	dl	label	option	datalist	h4	article	command	tbody				
link	noscript	q	var	sub	mark	bdi	wbr	figcaption	ul	dt	input	output	keygen	h5	footer	summary	thead				
style	script	cite	samp	sup	ruby	bdo	code	figure	li	dd	textarea	button	progress	h6	hgroup	details	tfoot				
										img	area	map	embed	object	param	source	iframe	canvas	track	audio	video

- Legend:
- Root element
 - Text-level semantics
 - Forms
 - Tabular data
 - Metadata and scripting
 - Grouping content
 - Document sections
 - Interactive elements
 - Embedding content

HTML5, une amélioration pour le référencement

1/2

HTML5 apporte une spécification des `<div>`, améliorant ainsi la sémantique du langage. Cette sémantique permet donc de mieux baliser son contenu et donc d'en améliorer le référencement.

- **section** : définit les sections dans un document. Tels que les chapitres, en-têtes, pieds de page, ou toutes autres sections du document.
- **article** : partie indépendante du site, comme un commentaire.
- **aside** : associé à la balise qui le précède.
- **header** : spécifie une introduction, ou un groupe d'éléments de navigation pour le document.

HTML5, une amélioration pour le référencement

2/2

- **footer** : définit le pied de page d'un article ou un document. Contient généralement le nom de l'auteur, la date à laquelle le document a été écrit et / ou ses coordonnées.
- **nav** : définit une section dans la navigation.
- **figure** : définit des images, des diagrammes, des photos, du code etc...
- **figcaption** : légende pour la balise `<figure>`.
- **details** : précise les détails supplémentaires qui peuvent être masqués ou affichés sur demande.

HTML5 une amélioration des formulaires

- Au delà de l'`<input>` :
 - **Les types avant** : *text, password, hidden, radio, checkbox, button, reset, submit, image & file.*
 - **Les type avec HTML5** : *tel, url, email, search, date, time, datetime, datetime-local, month, week, number, range & color.*
- Ouvert vers le web mobile grâce à ces spécification influençant l'IHM et améliorant l'UX.
- Amélioration de l'IHM en général avec des balises comme `<progress>` ou `<meter>`.
- Nouveaux attributs : *placeholder, autofocus, autocomplete, multiple* (et les implications qui vont avec !).

Amélioration au niveau des médias

Ordinateur

Mobile

Fonctionnalité	Chrome	Firefox (Gecko)	Internet Explorer	Opera	Safari
Support simple	3.0	3.5 (1.9.1)	9.0	10.50	3.1
<audio> : PCM en WAVE	(Oui)	3.5 (1.9.1)	Pas de support	10.50	3.1
<audio> : Vorbis en WebM	(Oui)	4.0 (2.0)	Pas de support	10.60	3.1 ^[1]
<audio> : Diffusion Vorbis/Opus en WebM via MSE	?	36.0 (36.0) ^[2]	?	?	?
<audio> : Vorbis en Ogg	(Oui)	3.5 (1.9.1)	Pas de support	10.50	Pas de support
<audio> : MP3	(Oui) ^[4]	(Oui) ^[5]	9.0	(Oui)	3.1
<audio> : MP3 en MP4	?	?	?	?	(Oui)
<audio> : AAC en MP4	(Oui) ^[6]	(Oui) ^[7]	9.0	(Oui)	3.1
<audio> : Opus en Ogg	27.0	15.0 (15.0)	?	?	?
<audio> : FLAC	56.0	51 (51)	Pas de support	Pas de support	Pas de support
<audio> : FLAC en Ogg	56.0	51 (51)	Pas de support	Pas de support	Pas de support
<video> : VP8 et Vorbis en WebM	6.0	4.0 (2.0)	9.0 ^[8]	10.60	3.1 ^[9]
<video> : VP9 et Opus en WebM	29.0	28.0 (28.0) ^[36]	?	(Oui)	?
<video> : Diffusion WebM via MSE	?	42.0 (42.0) ^[35]	?	?	?
<video> : Theora et Vorbis en Ogg	(Oui)	3.5 (1.9.1)	Pas de support	10.50	Pas de support
<video> : H.264 et MP3 en MP4	(Oui) ^[3]	(Oui) ^[10]	9.0	(Oui)	(Oui)
<video> : H.264 et AAC en MP4	(Oui) ^[4]	(Oui) ^[11]	9.0	(Oui)	3.1
<video> : FLAC en MP4	62	51 (51)	?	?	?
Tout autre format	Pas de support	Pas de support	Pas de support	Pas de support	3.1 ^[12]

Ordinateur

Mobile

Fonctionnalité	Android	Firefox Mobile (Gecko)	IE Mobile	Opera Mobile	Opera Mini	Safari Mobile	Chrome pour Android
Support simple	2.3	24.0	10.0	11.0	(Oui) ^[13]	3.2	29.0
<audio> : PCM en WAVE	?	24.0	Pas de support	Pas de support	(Oui) ^[14]	3.2	?
<audio> : Vorbis en WebM	?	24.0	Pas de support	11.0	(Oui) ^[15]	Pas de support	?
<audio> : Diffusion de Vorbis en WebM via MSE	?	?	?	?	?	?	?
<audio> : Vorbis en Ogg	?	24.0	Pas de support	11.0	(Oui) ^[16]	Pas de support	?
<audio> : MP3	?	(Oui) ^[17]	10.0	?	(Oui) ^[18]	3.2	?
<audio> : MP3 en MP4	?	?	?	?	?	(Oui)	?
<audio> : AAC en MP4	?	(Oui) ^[20]	10.0	?	(Oui) ^[22]	(Oui)	?
<audio> : Opus en Ogg	Pas de support	24.0	Pas de support	Pas de support	(Oui) ^[23]	Pas de support	Pas de support
<video> : VP8 et Vorbis en WebM	2.3	24.0	Pas de support	16.0	(Oui) ^[24]	Pas de support	29.0
<video> : VP9 et Opus en WebM	?	?	?	?	?	?	?
<video> : Diffusion de WebM via MSE	?	42.0 (42.0)	?	?	?	?	?
<video> : Theora et Vorbis en Ogg	Pas de support	24.0	Pas de support	Pas de support	(Oui) ^[25]	Pas de support	Pas de support
<video> : H.264 et MP3 en MP4	(Oui) ^[26]	24.0 ^[33]	10.0	16.0 ^[28]	(Oui) ^[29]	(Oui)	29.0
<video> : H.264 et AAC en MP4	(Oui) ^[30]	24.0 ^[34]	10.0	16.0 ^[28]	(Oui) ^[32]	3.2	29.0
<video> : FLAC en MP4	?	58.0 (58)	?	?	?	?	62.0
Tout autre format	?	?	?	?	?	?	?

- Implémente la balise <audio>.
- Implémente la balise <video>.
- Avec les nouveaux n

Autres améliorations

- Dessin avec la balise `<canvas>`,
- API de Géolocalisation,
- File API,
- Drag & Drop,
- Évènements envoyés par le serveur : push,
- Communication temps réel : Web socket,
- Stockage des données locales : Web Storage,
- Base de données,
- Le mode off-line.

Responsive Web Design

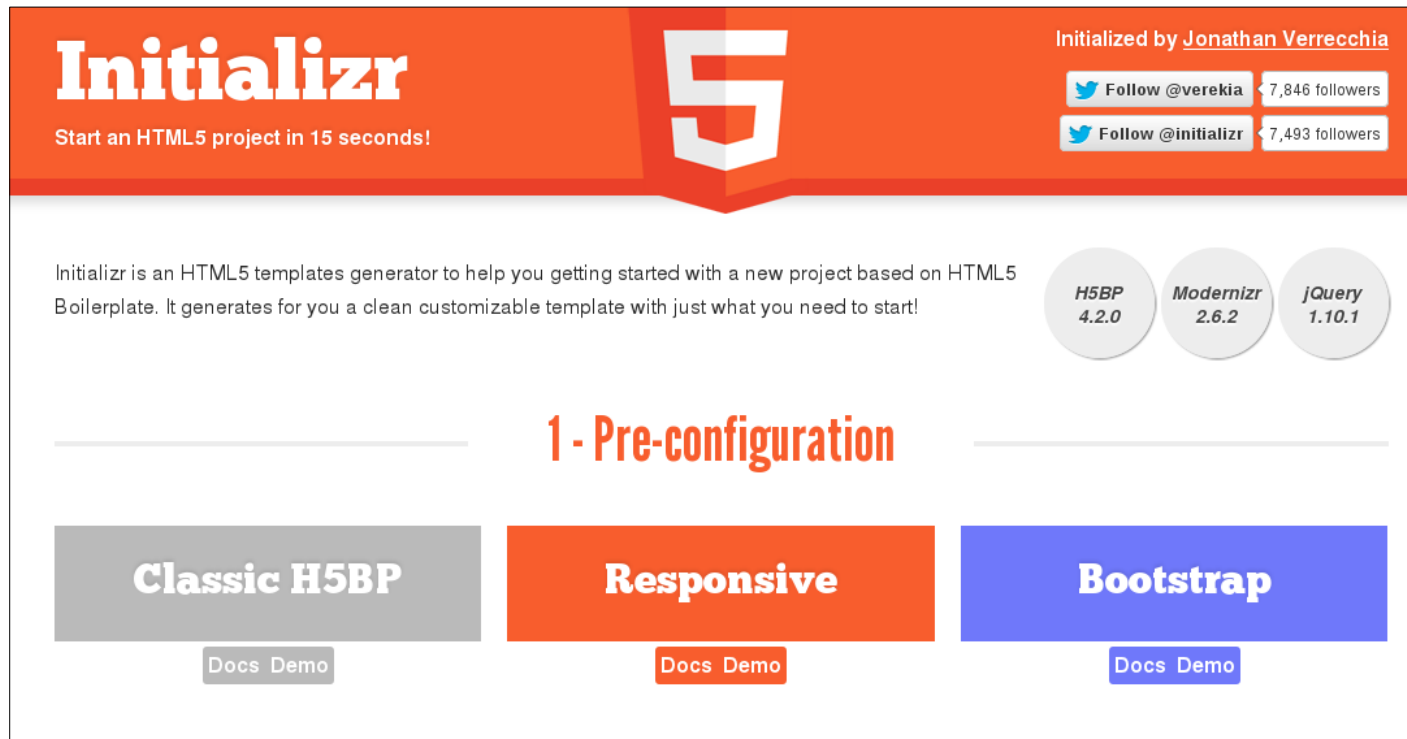
La notion de Responsive Web Design (conception de sites web adaptatifs selon l'OQLF) regroupe différents principes et technologies qui forment une approche de conception de sites Web dans laquelle un site est conçu pour offrir au visiteur une expérience de consultation optimale facilitant la lecture et la navigation.

L'utilisateur peut ainsi consulter le même site Web à travers une large gamme d'appareils (moniteurs d'ordinateur, smartphones, tablettes, TV, etc.) avec le même confort visuel et sans avoir recours au défilement horizontal ou au zoom avant/arrière sur les appareils tactiles notamment, manipulations qui dégradent considérablement l'expérience utilisateur.

Le Responsive Web Design est souvent abrégé par RWD dans les articles traitant du sujet. ©Wikipedia

Templates HTML5

- **Initializr** Fourni un générateur de template HTML5 tout prêt incluant la bonne structure ainsi que les scripts de détection et de modernisation.



The screenshot shows the Initializr website interface. At the top, there's an orange header with the 'Initializr' logo, a large '5' icon, and social media links for '@verekia' and '@initializr'. Below the header, a description states: 'Initializr is an HTML5 templates generator to help you getting started with a new project based on HTML5 Boilerplate. It generates for you a clean customizable template with just what you need to start!'. To the right of this text are three circular badges for 'H5BP 4.2.0', 'Modernizr 2.6.2', and 'jQuery 1.10.1'. The main section is titled '1 - Pre-configuration' and features three large buttons: 'Classic H5BP' (grey), 'Responsive' (orange), and 'Bootstrap' (blue). Each button has a 'Docs Demo' link below it.

Initializr
Start an HTML5 project in 15 seconds!

Initialized by [Jonathan Verrecchia](#)

[Follow @verekia](#) 7,846 followers
[Follow @initializr](#) 7,493 followers

Initializr is an HTML5 templates generator to help you getting started with a new project based on HTML5 Boilerplate. It generates for you a clean customizable template with just what you need to start!

H5BP 4.2.0 Modernizr 2.6.2 jQuery 1.10.1

1 - Pre-configuration

Classic H5BP **Responsive** **Bootstrap**

[Docs Demo](#) [Docs Demo](#) [Docs Demo](#)

2 - CSS 3

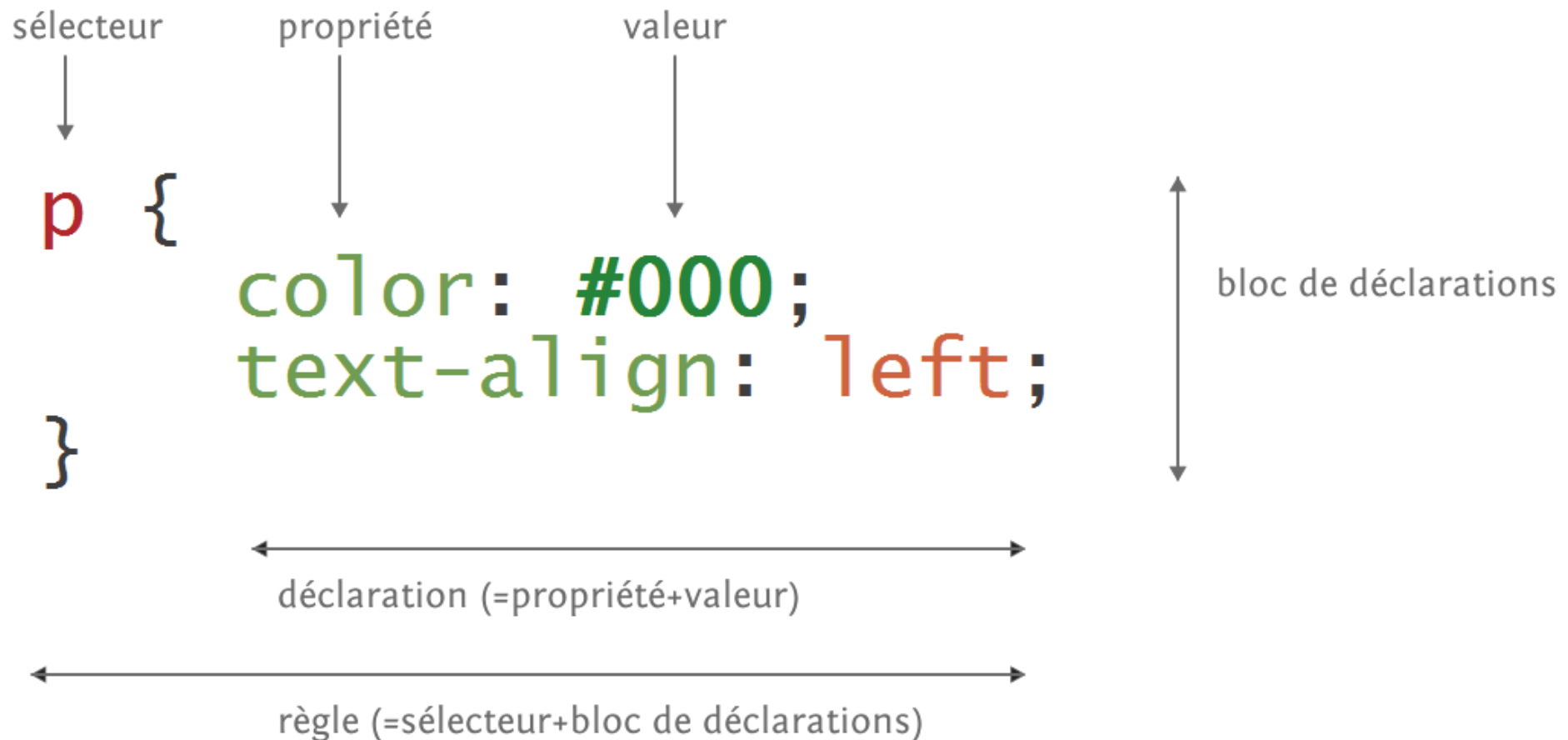
- Fonctionnement du CSS
- Les sélecteurs
- Les règles
- Les boîtes
- Le positionnement
- Frameworks

CSS c'est quoi ?

CSS (Cascading Style Sheets : feuilles de style en cascade) est un langage informatique qui sert à décrire **la présentation** des documents HTML et XML.

Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web dans les années 2000. © *Wikipedia*

Anatomie d'une règle CSS



© CSS avancées, vers HTML5 et CSS3.

Les sélecteurs principaux

- Sélecteurs de balise :

```
a {  
  text-decoration: underline  
}
```

- Sélecteurs de classe :

```
p.description {  
  color: #333333  
}
```

- Sélecteur d'identifiant :

```
p#sub-title {  
  color: #000;  
  font-weight: bold  
}
```

Les pseudo-classes & pseudo-éléments

- Pseudo-classes :
 - *:link*
 - *:visited*
 - *:hover*
 - *:focus*
 - *:active*
 - *:first-child*
 - etc.
- Pseudo-éléments :
 - *:first-letter*
 - *:first-line*
 - *:before*
 - *:after*,

Les sélecteurs particuliers

- Sélecteur d'enfant : ne s'applique qu'au niveau n-1

```
ul > li {  
  list-style : none ;  
}
```

- Sélecteur de frère adjacent :

```
h1 + p {  
  font-weight : bold  
}
```

- Sélecteur d'attribut

```
a.external[href^="https://"] {  
  font-style : italic;  
}
```

Sélecteur d'enfant

```
<p>Un paragraphe contenant  
  <a>un lien  
    <em>important</em>  
  </a>  
</p>
```

← parent <p>
← enfant <a>
← descendant

p > em {background: green;}	→	 n'est pas ciblé, pas de couleur
p em {background: red;}	→	 devient rouge
a > em {background: blue;}	→	 devient bleu
a em {background: yellow;}	→	 devient jaune

©CSS avancées, vers HTML5 et CSS3.

Sélecteur de frère adjacent

```
<h1>Un titre de niveau 1</h1>
```

← frère <h1>

```
<p>Un premier paragraphe</p>
```

← frère <p>

```
<p>Un second paragraphe</p>
```

← frère <p>

`h1 + p {background: #333;}` → seul le 1er paragraphe est ciblé

© CSS avancées, vers HTML5 et CSS3.

Poids des règles

Il est reconnu que lorsque deux règles CSS différentes ciblent le même élément, c'est la dernière déclarée (la plus basse dans le code source) qui s'applique et qui écrase la précédente. En pratique, cette loi ne vaut que dans le cas où les deux sélecteurs incriminés sont exactement du même type et déclarés de la même manière, car il existe une priorité entre les différents types de sélecteurs.

1. Règle CSS déclarée à l'aide de l'attribut HTML `style=` au sein de l'élément.
2. sélecteur d'identifiant (id).
3. Sélecteur de classe, d'attribut (`[]`) ou de pseudo-classe (`:hover`, `:focus`, `:first-child`).
4. Sélecteur d'élément (`p`, `div`, `a`,...) ou de pseudo-élément (`:first-letter`, `:before`, `:after`, `:first-line`).
5. Sélecteur joker (*), de parenté (>) ou d'adjacence (+).

Modèle de boîte

- Est appelé boîte l'espace occupé par un éléments HTML de type boîte (div, section, etc..) par opposition avec un éléments de type inline (a, span, etc..) qui s'insert dans le flux de texte.
- Un éléments inline peut être forcé en affichage boîte via CSS (display : block) et inversement (display : inline)



Organisation en flux

`<h1>`Titre de niveau 1

`<p>`Paragraphe

`<a>`lien

`<a>`lien

``image

``liste non ordonnée

``élément de liste

``élément de liste

Le positionnement en CSS 2.1 : position

<p>boîte 1</p>

<p>boîte 2</p>

<p>boîte 3</p>

Lorem Elsass ipsum réchime amet
bissame so **kiwi**. Leo Richard
Schirmeck tellus kartoffelsalad set

← élément #kiwi en
flux classique

Lorem Elsass ipsum réchime amet
bissame so **kiwi** Leo Richard
Schirmeck tellus kartoffelsalad set

← élément #kiwi
décalé avec
position: relative;
top: 10px;
left: 10px;

© CSS avancées, vers HTML5 et CSS3.

Le positionnement en CSS 2.1 : float

<p> Paragraphe flottant | <p> Autre paragraphe en flux. Lorem Elsass ipsum réchime amet bissame so libero. Schirmeck tellus kartoffelsalad set. Hopla !

bloc parent

© CSS avancées, vers HTML5 et CSS3.

Display : flex

- **Direction**

```
.container {
```

```
    flex-direction: row | row-reverse | column | column-reverse;
```

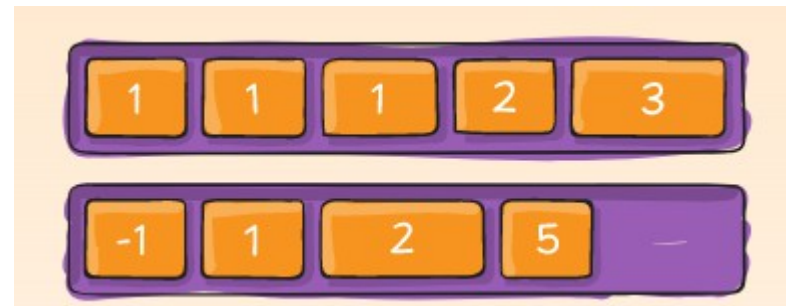
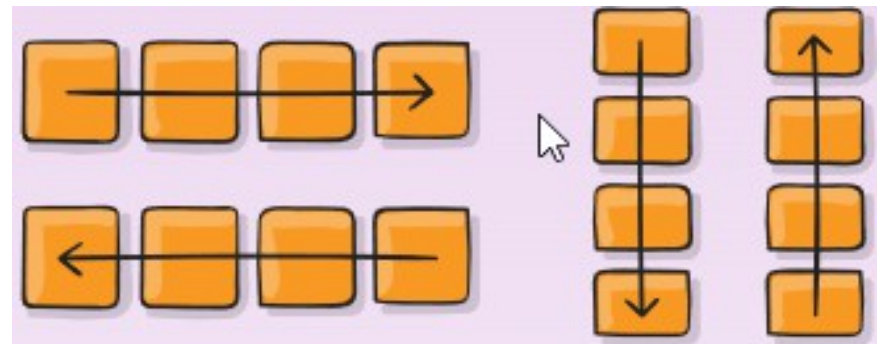
```
}
```

- **Ordre**

```
.item {
```

```
    order : 5 /* Default is 0 */
```

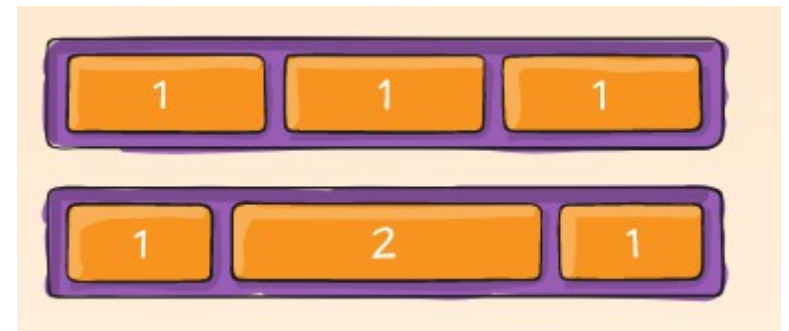
```
}
```



Display : flex

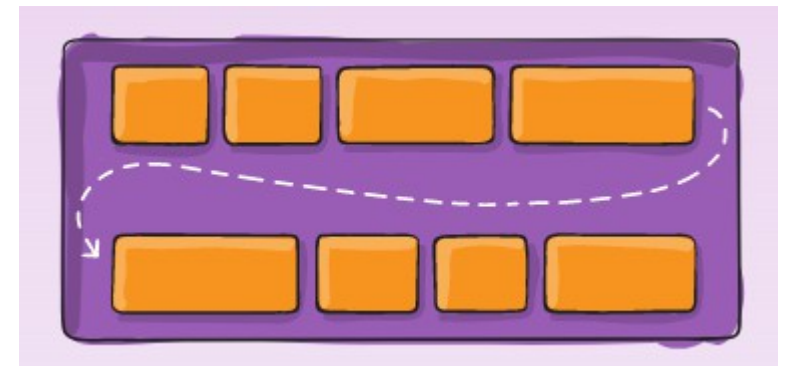
- **Taille des éléments**

```
.item {  
    flex-grow: 4;  
}
```



- **Passage à la ligne**

```
.container {  
    flex-wrap: nowrap | wrap | wrap-reverse  
}
```

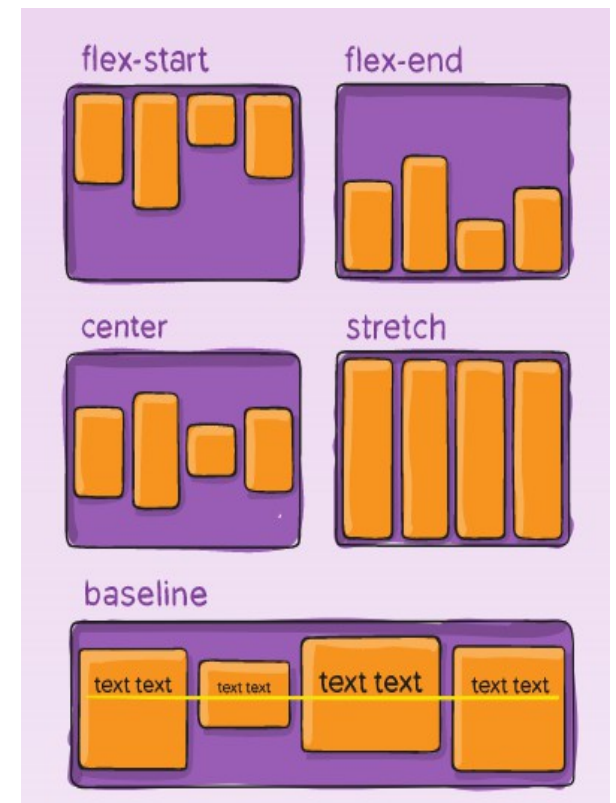
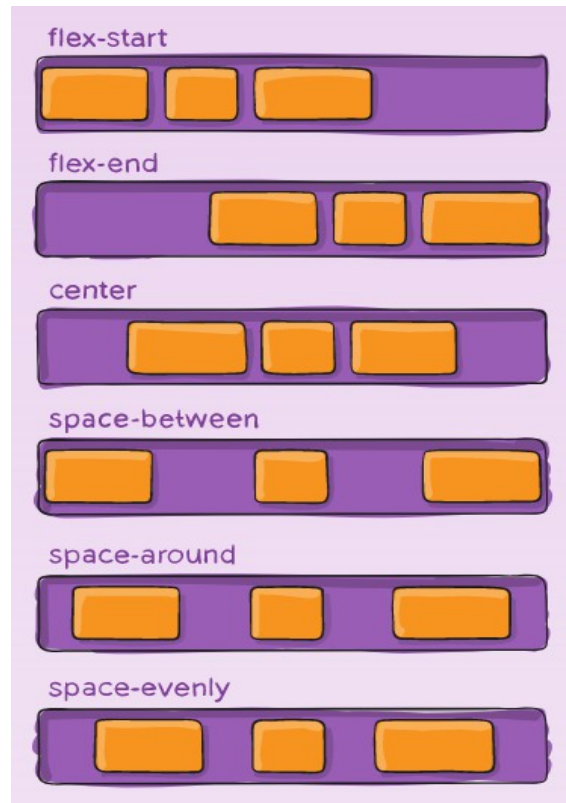


Display : flex

- Alignements :**

```
.container {
    justify-content : flex-start |
    flex-end | center | space-between | ... ;

    align-items : stretch | flex-start
    | flex-end | center | start | end | ... ;
}
```



Dans la pratique

- Dans la pratique, je conseille de privilégier le modèle en flex qui offre l'avantage d'être compatible avec une grande partie des navigateurs.
- Les méthodes avancées peuvent être utilisées pour le Responsive Web Design.
- [Excellente documentation sur les Flex Box.](https://css-tricks.com/snippets/css/a-guide-to-flexbox/)

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

RWD & Media Queries

- Approche historique :

```
<link rel="stylesheet" media="screen" href="screen.css"
type="text/css" />
<link rel="stylesheet" media="print" href="print.css"
type="text/css" />
```

- Syntaxes CSS3 :

```
@media screen and (min-width: 200px) and (max-width: 640px) {
  .bloc {
    display: block;
    clear: both;
  }
}
```

Les préprocesseurs CSS

- Les préprocesseurs CSS sont des outils permettant d'utiliser une syntaxe plus riche que le CSS et de compiler ensuite une feuille CSS.
- Les plus connus sont **LESS** & **SAAS**.

```
// LESS                                     /* Compiled CSS */

@color: #4D926F;                             #header {
#header {                                     color: #4D926F;
  color: @color;                             }
}                                             h2 {
h2 {                                         color: #4D926F;
  color: @color;                             }
}
```

Pourquoi un framework ?

36

- Un bon développeur est un développeur fainéant !
- La roue existe déjà pourquoi la réinventer ?

Frameworks CSS

- Il existe un grand nombre de frameworks CSS offrant des styles par défaut pour certains éléments (formulaire, boutons, etc.). Attention cependant, ces styles sont là pour être personnalisés ! Malheureusement on constate trop de sites utilisant la feuille par défaut du framework, entraînant ainsi une uniformisation déplaisante du web.
- Parmi les Frameworks CSS notables :
 - [Bootstrap](#) by twitter qui est l'un des frameworks les populaires actuellement.
 - [Pure](#) by Yahoo ! Qui gagne à être connu.
 - [Blueprint](#) : l'un des plus anciens.
 - [Unsemantic](#), l'évolution 1180px / RWD de [960.gs](#), l'une des grilles les plus populaires.
- Dans tous les cas, si vous ne voulez pas utiliser de framework, je ne saurais vous conseiller d'utiliser un [reset.css](#).

3 - La performance du Web et les bonnes pratiques

- Pourquoi ?
- Minification
- Les bonnes pratiques

Pourquoi ? L'impact des performances sur le trafic

- « Perdre 500ms c'est perdre 20% de trafic pour Google (ou pourquoi il n'y a que dix résultats par page dans les recherches). »
- « Augmenter la latence de 100ms c'est perdre 1% de ventes pour Amazon. »
- « Réduire de 25% le poids de la page c'est gagner 25% d'utilisateurs à moyen terme pour Google. »
- « Perdre 400ms c'est avoir 5 à 9% d'abandons en plus pour Yahoo! sur un site éditorial. »

La performance influence le business !

- La minification consiste à alléger le code d'une feuille de style ou d'un scripts JS allant jusqu'à l'offusquer.
- Minification CSS :
 - Unification,
 - Retrait des commentaires,
 - Retrait des caractères blancs.
- Minification JS :
 - Pareil que CSS,
 - Peut aller jusqu'à renommer les variables pour les rendre plus courtes.
 - Peut aller jusqu'à faire du pack (~ JS compilé).
- Ce n'est pas parce qu'un script tourne non minifié qu'il tournera aussi une fois minifié. Il faut tester vos scripts minifiés !
- [YUI Compressor](#) & Google [Closure Compiler](#) étaient des références dans le domaine de la minification. Replacé par des outils NodeJS ([uglify-js](#), [uglify-css](#)).
- Il existe également des scripts pour minifier à la volé (attention au cache pour ne pas faire plus de mal que de bien !).

Optimiser votre code et votre site : les bonnes pratiques

41

- Minimize HTTP Requests
- Use a Content Delivery Network
- Avoid empty src or href
- Add an Expires or a Cache-Control Header
- Gzip Components
- Put StyleSheets at the Top
- Put Scripts at the Bottom
- Avoid CSS Expressions
- Make JavaScript and CSS External
- Reduce DNS Lookups
- Minify JavaScript and CSS
- Avoid Redirects
- Remove Duplicate Scripts
- Configure Etags
- Make AJAX Cacheable
- Use GET for AJAX Requests
- Reduce the Number of DOM Elements
- No 404s
- Reduce Cookie Size
- Use Cookie-Free Domains for Components
- Avoid Filters
- Do Not Scale Images in HTML
- Make favicon.ico Small and Cacheable

4 - Introduction à jQuery

- L'AJAX
- \$, l'objet jQuery
- Sélecteurs
- Manipulations

Rappel : pourquoi un framework ?

43

- Un bon développeur est un développeur fainéant !
- La roue existe déjà pourquoi la réinventer ?
- Le slogan jQuery : « Write less, do more »

AJAX, AJAJ, AJAH ???

- L'AJAX n'est pas une nouvelle technologie mais un ensemble de vieilles technologies (XMLHttpRequest, JS, HTML, etc...) saupoudré de magie commerciale (Web 2.0, Web 3.0, etc...)
- On parle généralement d'AJAX mais au final c'est plus de l'AJAH ou de l'AJAJ
- jQuery est riche en fonction AJAX :
 - `$.get()` ou `$(element).load()` (HTML).
 - `$.getJSON()` (JSON).
 - `$.getScript()`.

Le format JSON

- JSON pour **J**avaScript **O**bject **N**otation
- C'est une sérialisation d'objets
- Format léger et séparant la présentation du contenu
- Compatible PHP, JavaScript, Python, etc...

```
{ "menu": {  
  "id": "file",  
  "value": "File",  
  "popup": {  
    "menuitem": [  
      { "value": "New", "onclick": "CreateNewDoc()" },  
      { "value": "Open", "onclick": "OpenDoc()" },  
      { "value": "Close", "onclick": "CloseDoc()" }  
    ]  
  }  
}
```

jQuery et « \$ »

- Le symbole « \$ » est un raccourci pour l'objet jQuery.
- Attention, ce symbole « \$ » est utilisé par d'autres framework javascript. Si vous utilisez différents framework vous devrez forcer l'appel à jQuery à chaque fois ou utiliser `jQuery.noConflict();`

Sélection des portions du DOM

- La force de jQuery est de facilement pouvoir récupérer des informations issues du DOM :

// Selecting Elements by ID

`$("#myId");` // Note IDs must be unique per page.

// Selecting Elements by Class Name

`$(".myClass");`

//Selecting Elements by Attribute

`$("input[name='first_name']");`

// Selecting Elements by Compound CSS Selector

`$("#contents ul.people li");`

- jQuery autorise même l'utilisation de pseudo-sélecteur :
<http://learn.jquery.com/using-jquery-core/selecting-elements/>

Manipuler le DOM

- Une fois le DOM récupéré, on peut le manipuler :

```
// The .html() method used as a setter:  
$( "h1" ).html( "hello world" );
```

```
// The .html() method used as a getter:  
$( "h1" ).html();
```

```
// Attempting to call a jQuery method after calling a getter.  
// This will NOT work:  
$( "h1" ).html().addClass( "test" );
```

- Les possibilités sont énormes, on peut par exemple :
 - Inverser l'ordre des éléments dans un DOM,
 - Supprimer des portions de DOM,
 - Transformer des éléments,
 - En rajouter,
 - Etc.

Le chainage et les appels plus complexes

Voici un exemple de sélection / manipulation plus complexe et chaîné :

```
$( "#content" )  
  .find( "h3" )  
  .eq( 2 ) // Select the third h3  
  .html( "new text for the third h3!" )  
  .end() // Restores the selection to all h3s  
  .eq( 0 ) // Select the first h3  
  .html( "new text for the first h3!" );
```

Les événements

- Afin de dynamiser une page, on doit pouvoir y déposer des événements entraînant des actions :
 - Actions au clic,
 - Actions lorsque l'on quitte le champ,
 - Etc.

```
// The hover helper function
$( "#menu li" ).hover(function() {
    $( this ).toggleClass( "hover" );
});
```

```
// Event setup using a convenience method
$( "p" ).click(function() {
    console.log( "You clicked a paragraph!" );
});
```

Les effets

- Une autre façon d'agrémenter une page web et d'y rajouter des effets afin d'enrichir l'expérience utilisateur (UX):
 - Apparition progressive de message
 - Transition progressive au sein de la page
 - Etc.

```
// Slowly hide all paragraphs  
$( "p" ).hide( "slow" );
```

```
// Quickly show all divs that have the hidden style class  
$( "div.hidden" ).show( "fast" );
```

Création de plugins jQuery

- JQuery offre un grand nombre de fonctions mais il est possible d'enrichir le langage via des plugins :

```
/**
 * Load brdwelcome with AJAX
 *
 * <code>
 * $("#brdwelcome").fedoraBrdwelcome();
 * </code>
 */
$.fn.fedoraBrdwelcome = function ()
{
    if ( this.length && this.hasClass('ajax'))
    {
        this.load( $.ez.ezroot + 'ajax/brdwelcome.php');
    }
}
```

```
/**
 * Action to make on document
ready
*/
$(document).ready
(function(){
    // brdwelcome
    $
    ('#brdwelcome').fedoraBrdwelco
me();
});
```

Les plugins sympas

- [jQuery Table sorter](#) : pour trier dynamiquement des tableaux.
- [Topsy](#) : des ToolTip à la Facebook.
- [Full calendar](#) : représentation en calendrier d'une liste d'événements en JSON.
- [ColorBox](#) : Lightbox en jQuery.
- [Validation](#) : validation de formulaires.

5 - Sémantique Web et SEO

- La définition du SEO
- L'importance de Google
- L'importance du contenu
- Panda & Pingouin

Le SEO c'est quoi ?

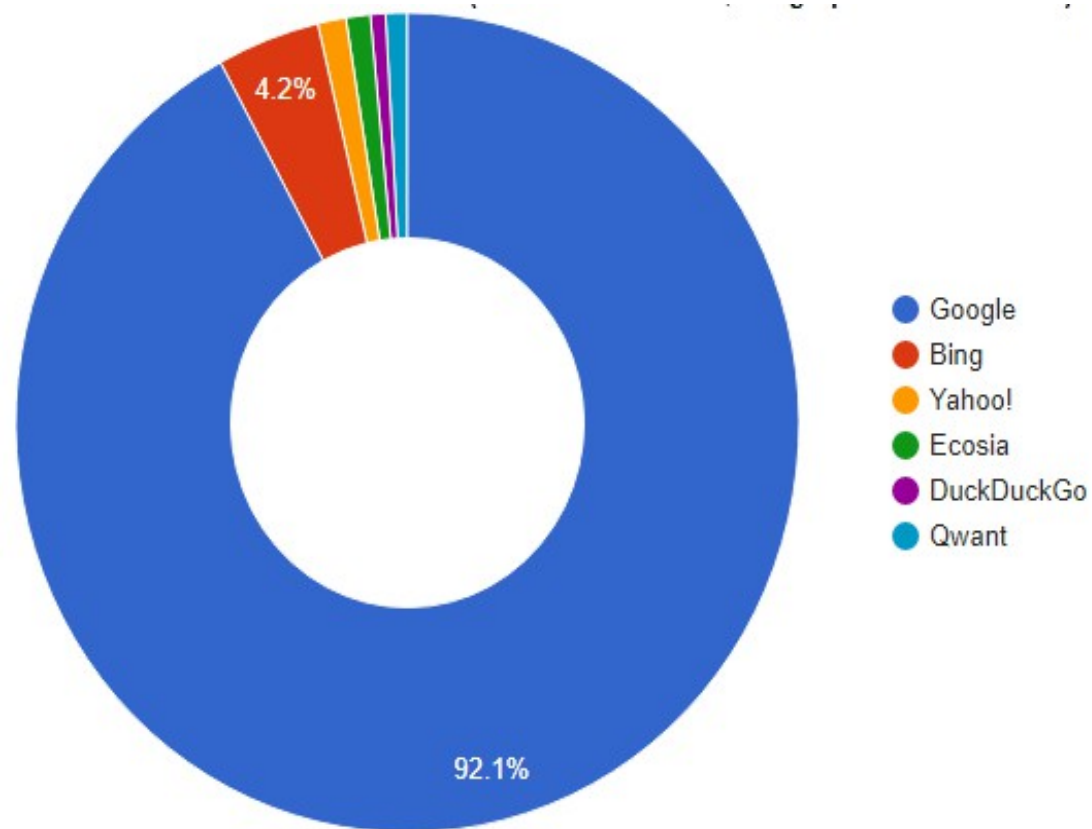
L'**optimisation** pour les moteurs de recherche (en anglais, **Search engine optimization : SEO**) est un ensemble de **techniques** visant à favoriser la **compréhension** de la thématique et du contenu d'une ou de l'ensemble des pages d'un site Web par les moteurs de recherche. Ces techniques visent donc à apporter un maximum **d'informations** concernant le contenu d'une page web aux robots d'indexation des moteurs de recherche. Ces robots, en arrivant sur le site, laissent une empreinte sur le serveur dans le fichier log. Leur passage est repérable par une « signature » propre à chacun.

L'objectif de ce procédé est **d'orienter** le **positionnement** d'une page Web dans les résultats de recherche des moteurs sur des mots-clés correspondant aux thèmes principaux du site. On considère généralement que le positionnement d'un site est bon lorsqu'il est positionné (classé) dans **l'une des dix premières réponses** d'une recherche sur des mots-clés correspondant précisément à sa thématique. L'optimisation pour les moteurs de recherche est un des métiers de l'internet. (© *Wikipedia*)

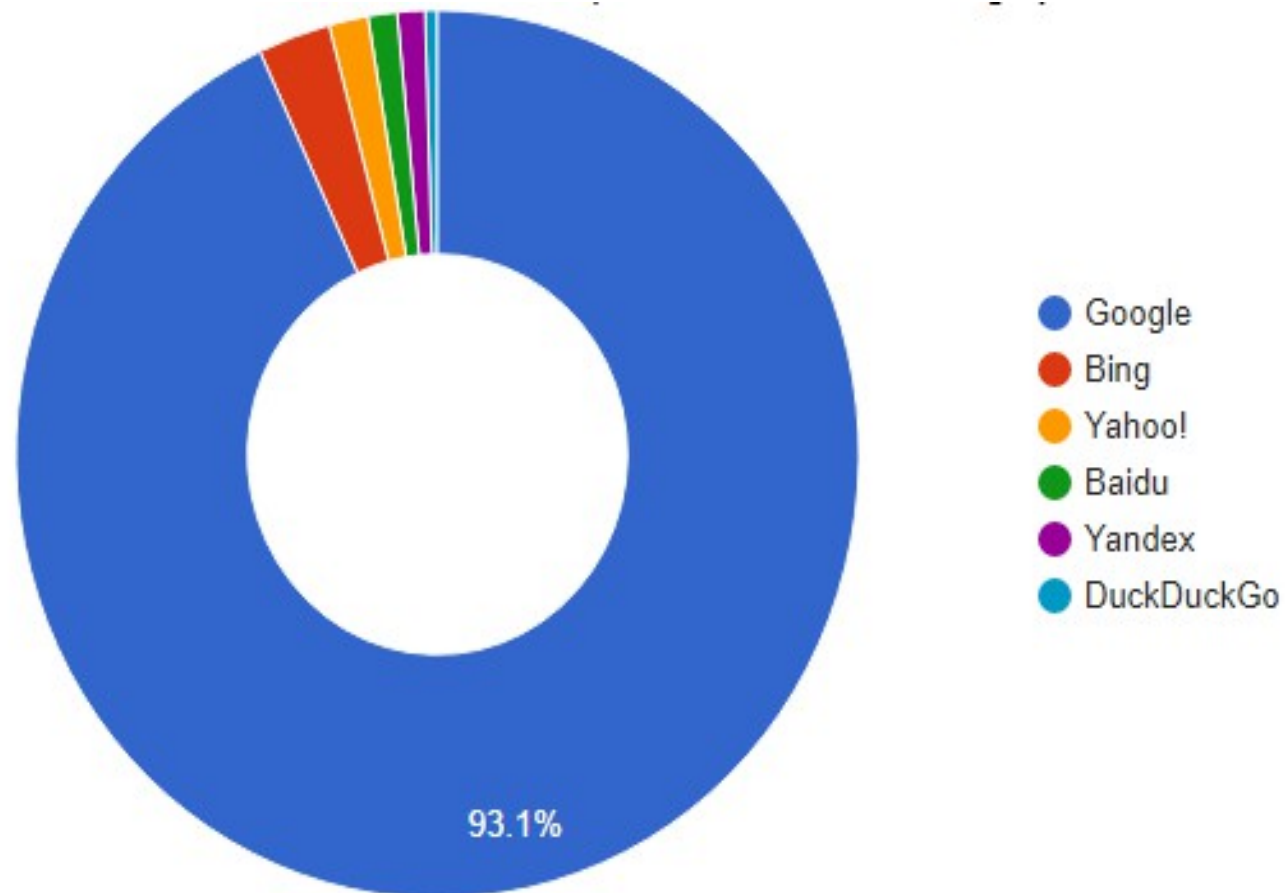
L'importance d'un bon référencement

- Être visible sur son nom
- Être visible sur son business
- Masquer les avis négatifs et avoir bonne réputation.

La part des moteurs de recherche en France



La part des moteurs de recherche dans le monde



Le contenu avant tout

- Ce que Google indexe c'est du **contenu** !
- Une page avec moins de 100 à 200 mots descriptifs dans son contenu éditorial est reversée dans l'index secondaire.
- Si vous n'avez pas de contenu en relation avec les mots pour lesquels vous voulez être référencé vous ne pourrez y être référencé.

SEO et anthropologie

- Pas de règles écrites.
- Science basée sur l'observation.
- Modifications du comportement du sujet dues à son observation.
- Certains points sont marqués « *divergence* » dans les slides. C'est qu'il existe des divergences d'opinions sur ces règles.

Les métas

Bien qu'en forte baisse, ne pas négliger les balises métas :

- Title (divergences sur la structure),
- Description,
- Keywords,
- Robots.

Le nom de domaine

- Doit contenir des mots clefs et des mots qui existent.
- Confiance à l'ancienneté.
- Redirection des noms de domaine secondaires (*duplicate content*).

KO

<http://event-2011.monsite.fr>

<http://dexter-saison2.monsite.fr>

OK

<http://event.monsite.fr/2011>

<http://dexter.monsite.fr/saison2>

Format de l'URL

- Bannir les caractères tels que « * », « + » ou « ? ».
- Minuscules sans accents (*divergences*).
- Séparer avec des « - » mais pas de « _ ».

KO

http://www.monsite.fr/événement_2011

OK

<http://event.monsite.fr/evenement/2011>

http://www.monsite.fr/event_2011

<http://www.monsite.fr/event-2011>

Les URLs

- Une URL doit toujours être accessible (*archive*).
- Unicité des URLs (produit dans différentes catégories) et du contenu (*duplicate content*).
- Ne pas polluer les URLs avec les sessions.

La sémantique web

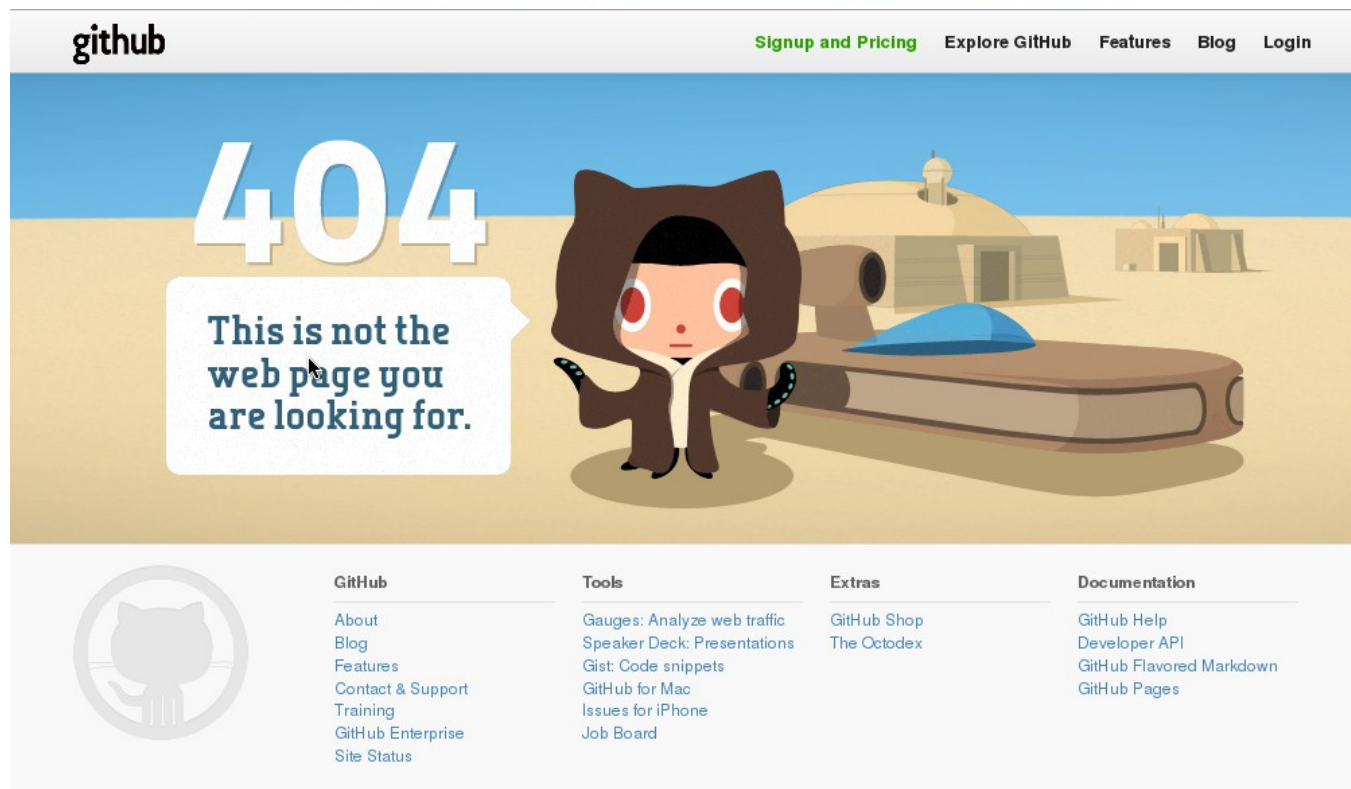
Utiliser les balises HTML pour leur fonction première et ne pas les détourner pour de la présentation :

- Utiliser des structure de `<hx>`.
- Utiliser les ``.
- Attribut « alt » dans les balises images.
- Listes ``, ``, `<dl>`, etc...
- Tableaux pour des données tabulaires.


Les liens

- Le lien est un critère important pour le référencement de la **page cible** !
- Ne pas utiliser de titres communs tels que « cliquer ici », « Lire la suite », etc...

404




Cross content & cross linking


Guillaume Kulakowski's blog

Accueil
Abonnements
Archives
Mots-clés
CV

Parmi les nombreuses nouveautés, on remarquera :


- Des corrections de bugs en tout genre.
- Le support de nouveaux périphériques.
- L'identifiant unique d'un équipement Jeedom (LogicalId) passe de l'adresse (ADDR) à l'IEEE. L'idée et d'assurer un identifiant unique qui ne change pas au grès des rafraichissements de l'équipement.
- Un support complet des firmware 3.0e et 3.0f. Il était déjà opérationnel pour les nouvelles installations mais la version 1.2.0 introduit un script de migration de l'addr vers l'IEEE pour les anciennes installations.
- L'arrivée d'une console permettant d'envoyer des commandes directement à votre ZiGate.




La console permettant de lancer des commandes vers votre ZiGate directement depuis Jeedom.

La liste complète des changements est disponible [ici](#) et votre gestionnaire de mise à jour de Jeedom devrait vous proposer la mise à jour très prochainement.


Articles similaires




Mon installation domotique
Dans "Domotique"



Jeedom, installation / update sous Odroid
Dans "Domotique"




Migration vers la ZiGate de mes équipements Xiaomi
Dans "Domotique"



MyMuleSoftSummit2019
Conséquence de l'effet heureux et hasarde...
[linkedin.com](#)

3 oct. 2019

Guillaume Kulakowski a retweeté







Gacina Pascal
@GacinaP
#cgi présent au #MulesoftSummit2019

Benoît Pradelles, @llaumgui#shapsha
#digitaltransformation


Intégrer
Voir sur Twitter

Navigation








Mon installation domotique
13 mai 2018 Domotique



Docker pour ma stack LAMP
25 avril 2018 Sysadmin



Grunt, Bower, LESS & Bootstrap
24 janvier 2014 Web & bonnes pratiques

Divers

- Importance des commentaires : synonymes, orthographe, SMS, féminin, pluriel, etc...
- robots.txt.
- Sitemaps.
- Plan du site.
- Backs links (home + interne)

Recherche à l'intérieur de votre site

- Un bon moteur de recherche permet de conserver vos visiteurs sur votre site.
- Importance de s'y retrouver dans un contenu vaste.
- Une base de données est faite pour stocker de la donnée, elle n'est pas faite pour y rechercher de la donnée !
- Utilisation de **facets** pour filtrer les résultats de recherches.
- « *Spelling search* » où la recherche par approximation orthographique.
- Notion de pertinence.

Les solutions open sources de recherches

- **Solr / ElasticSearch**: Solution en Java qui est utilisée sur Wikipédia.
- **Sphinx** : Solution en C, utilisée sur un grand nombre de board warez.
- Des principes identiques :
 - indexer le contenu afin de construire un index.
 - Ne permet pas d'afficher un contenu mais retourne un indentifiant + un highlight.
 - Scoring
 - Élévation

6 – TP n°1

- Exercices jQuery

- Téléchargez le fichier « TP1.tar.gz ».
- Décompressez l'archive.
- Lancer le fichier index.html.
- Faites ce qui est demandé, les différents fichiers sont là pour vous aider.

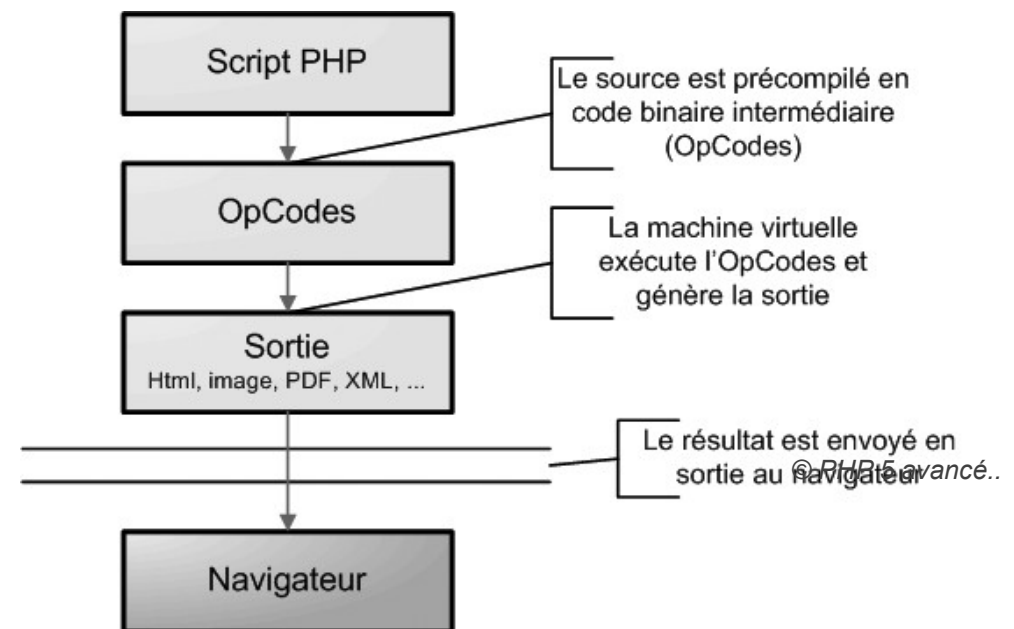
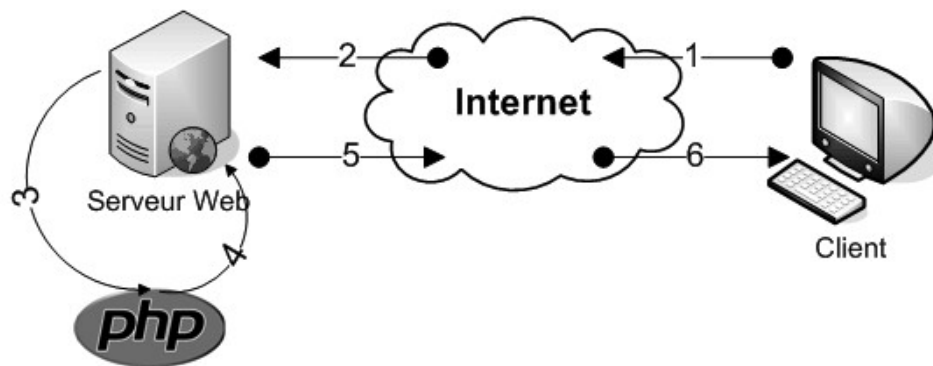
7 - Présentation du langage PHP

- Les fondamentaux
- Le fonctionnement
- L'histoire
- PHP 6

PHP en quelques mots

- Un langage Open Source : la Licence PHP **est approuvée par la FSF mais pas GPL compatible**, pas de frais de licence,
- Exécution via « *byte code* » recréé à chaque exécution (sauf si cache OPCode) pas de compilation,
- Langage à la prise en main rapide,
- Langage extensible (PECL, PEAR, PEAR2, Composer),
- Langage pour :
 - Applications serveur (sites web),
 - Scripts,
 - Web services,
 - Applications graphiques (php GTK, mais à déconseiller !),
- Langage à typage faible mais typable (de + en + typable avec PHP 7),
- Performant et fiable n'en déplaie à ses détracteurs. Le problème de php est d'être trop simple et de permettre à des personnes incompetentes de distribuer du code à la qualité / sécurité discutable !

Fonctionnement de PHP



L'histoire de PHP 1/3

- **La genèse de PHP :**
 - Créé en 1994 par Rasmus Lerdorf,
 - Bibliothèque en Perl,
 - Connaitre la consultation du CV de l'auteur,
 - **Personnal Home Page Tools.**
- **PHP/FI 1.0 :**
 - **Personal Home Page / Form Interpreter,**
 - Libération du code en 1995,
 - Migration de Perl en C,
 - 1997 : 1% des noms de domaines.
- **PHP/FI 2.0 :**
 - Novembre 1997.

L'histoire de PHP 2/3

- **PHP 3.0 :**
 - Libération en 1998,
 - Refonte de PHP/FI plus qu'une suite,
 - Refonte qui fait suite au développement d'une application eCommerce qui rencontrait des problèmes de performances,
 - Par **Andi** Gustman et **Zeev** Suraski (fondateurs de Zend),
 - Première vraie version suffisamment stable pour la production,
 - Rasmus rejoint PHP qui devient donc la suite de PHP/FI,
 - **PHP: Hypertext Preprocess**,
 - Ajout du support des extensions,
 - 10 % du parc serveur.

L'histoire de PHP 3/3

- **PHP 4.0 :**

- Libération en 2000,
- Réécriture totale du moteur pour améliorer les performances,
- Naissance du Zend Engine :
 - Analyseur syntaxique,
 - Procède en deux étapes : analyse puis exécution.

- **PHP 5.0 :**

- Libération en 2004,
- Simplification & professionnalisation,
- Zend Engine II avec intégration complète de la POO,
- Refonte du moteur XML,
- SQLite,
- PDO.

L'après PHP 6, PHP 5.3+

- **PHP 5.3 :**
 - Support des espaces de nom,
 - Phar,
 - Closure,
 - GC.
- **PHP 5.4 :**
 - Trait,
 - Serveur HTTP.
- **PHP 5.5 :**
 - Libération de Zend OPcache,
 - Support mysql obsolète préférer mysqlnd ou mysqli.
- **PHP 5.6 :**
 - Nouveau déboguer.

L'après PHP 6, PHP 7

- **PHP 7 :**

- Amélioration des performances (x2).
- Conversion d'un grand nombre de « fatal errors » en « Exceptions ».
- Typage des retours.
- Support des scalaires dans les déclarations.
- Classes anonymes.

- **PHP 7.1 :**

- Pseudo-type "Iterable"
- Capture d'exceptions multiple
- Visibilité des constantes de classe
- Type de retour "void"
- Types "nullable"

L'après PHP 6, PHP 7

- **PHP 7.2 :**

- Type hint (déclarations de type d'argument) de type object.
- Déclarations return type de l'objet.
- Type de paramètre Widening.
- Virgules de fin dans la syntaxe de liste.
- Argon2 dans le hachage du mot de passe.

- **PHP 7.3 :**

- Implémentation des syntaxes flexibles Heredoc et Nowdoc.
- Autorisation des virgules dans les appels d'une fonction.
- JSON_THROW_ON_ERROR.
- list() Affectation de référence.
- Fonction is_countable.
- array_key_first(), array_key_last().
- Améliorations apportées à Argon2 Password Hash.

L'après PHP 6, PHP 7

- **PHP 7.4 :**
 - Typed Properties
 - Arrow Functions
 - Limited Return Type Covariance and Argument Type Contravariance
 - Unpacking Inside Arrays
 - Numeric Literal Separator
 - Weak References
 - Allow Exceptions from `__toString()`
 - Opcache Preloading
 - Several Deprecations
 - Extensions Removed from the Core

L'arrivée de PHP 8

- **PHP 8**
 - Compilateur JIT
 - Typage plus fort (type union, Type errors, Type annotations)
 - Stringable interface
 - Simplification des constructeurs
 - Et bien d'autre à venir ...

8 – Installation & configuration

- LAMP
- WAMP
- MAMP

Sous Linux

- Plusieurs serveurs Web peuvent exécuter du PHP :
 - **Apache**,
 - Nginx,
 - PHP.
 - Etc...
- Plusieurs versions de php :
 - **mod_php** pour apache,
 - php_cgi,
 - php_fcgi,
 - **php_fpm**.
- Compatible avec plusieurs SGBD :
 - MySQL / **MariaDB**,
 - SQLite,
 - PostgreSQL,
 - Oracle,
 - Etc...
- Via le gestionnaire de paquets (yum, apt, etc...).

Sous Windows

- Pour le DEV et uniquement pour le DEV !
 - **WAMP**,
 - EasyPHP,
 - XAMPP,
 - Etc...
- Pour la production :
 - Utiliser un serveur Linux si possible...
 - Sinon, installation / configuration à la main.

Sous Mac

- Pour le DEV et uniquement pour le DEV !
 - [MAMP](#).
- Pour la production :
 - Utiliser un serveur Linux si possible...
 - Sinon, installation / configuration à la main.

En développement

- Configurer PHP pour afficher les erreurs :

```
display_errors = On  
display_startup_errors = On  
error_reporting = E_ALL  
html_errors = On
```

- Activer et user et abuser d'[Xdebug](#) :

```
xdebug.default_enable = 1  
xdebug.max_nesting_level = 200
```

9 – Syntaxe PHP

- Les bases
- PHP au sein d'HTML
- Hello World
- Commentaires
- Variables

Les bases

- Fichier .php (.php4 ou .php5 possible),
- Commence par <?php (Syntaxe <% et <? à proscrire),
- Se termine par ?> ou par rien si en fin de fichier,
- Ouverture / fermeture à répétition possibles, conjointement à HTML par exemple,
- « ; » après chaque instruction,
- Si erreur de syntaxe : « *Parse error* ».
- S'exécute en ligne de commande ou à travers le navigateur (<http://localhost>).

La syntaxe dans HTML

```
<html>
<head>
  <title><?php echo 'Titre'; ?></title>
</head>
<body>
  <h1>Titre niveau 1</h1>
  <?php
  echo "<p>ceci est du code PHP</p>";
  echo '<p>Là aussi !</p>';
  ?>
</body>
</html>
```

Hello world

```
<?php
// forme la plus simple, recommandée
echo 'Hello World';
?>
```

```
<?php
echo 'Hello ', 'World';
?>
```

```
<?='Hello World';
```

```
<?php echo 'Hello World'; echo 'Comment allez-vous ?'; echo 'Il fait
beau non ?' ?>
```

Les commentaires

- « // » ou « # » (déprécié)

// Commentaire sur une ligne

- « /* */ »

/* Commentaire
Sur plusieurs lignes
*/

/**
* Commentaire multi lignes avec phpDoc
*
* @param \$username String: username.
* @return bool
*/

Les variables

Correct	Incorrect	Explications
<code>\$variable1</code>	<code>\$Variable 1</code>	Contient des espaces
<code>\$variable</code>	<code>variable</code>	Une variable commence toujours par \$
<code>\$variable_double</code>	<code>\$variable-double</code>	Le signe - est interdit
<code>\$variable_email</code>	<code>\$test@yahoo.fr</code>	Les caractères @ et . sont interdits.
<code>\$test2</code>	<code>\$2test</code>	Une variable ne commence pas par un chiffre.

```
<?php
```

```
$variableFirst = 'Ma première variable, je suis sensible à la casse & je  
ne suis pas typée ! Ma portée est locale !' ;
```

Jouer avec les variables

- Une variable est par défaut locale sauf si :
 - `$GLOBALS['maVariable']`
 - `global $maVariable`
- On peut tester si une variable est définie avec `isset()` et détruire avec `unset()` :

```
$toTest = 'maVariable';  
if (isset($$toTest) === true) {  
    unset($maVariable) ;  
}
```


Les constantes

- PHP possède de vraies constantes :

`define('MA_CONSTANTE', 'En générale les constantes sont toutes en majuscule avec des « _ »');`

- On peut tester la définition d'une constante :

```
if (!defined('ROOT_PATH')) {  
    define('ROOT_PATH', dirname(__FILE__));  
}
```

Les tableaux

- Peut être défini via `array()` :

```
$myArray = array(1, 2, 3, 4, 5);
```

```
$mySugarArray = [1, 2, 3, 4, 5];
```

- Même fonction pour les tableaux et les tableaux associatifs :

```
array('nom' => 'Kulakowski', 'prenom' => 'Guillaume') ;
```

- Peut-être initié puis affecté au besoin :

```
$myArray = array() ;
```

```
$myArray[] = 'Guillaume' ;
```

```
$myArray['prenom'] = 'Guillaume';
```

- Peuvent-être manipulés avec **tout un tas de fonctions**.

Les opérateurs d'affectation, d'arithmétiques et de concaténations

Opérateur	Opération	Exemple
=	Affectation	<code>\$i = 1</code>
+	Addition	<code>echo \$a + \$b</code>
-	Soustraction	<code>echo \$a - \$b</code>
*	Multiplication	<code>echo \$a * \$b</code>
/	Division	<code>echo \$a / \$b</code>
%		<code>echo \$a % \$b</code>
++	Incrémentation	<code>\$i++</code> ou <code>++\$i</code>
--	Décrémentation	<code>\$i--</code> ou <code>--\$i</code>
X=	Opération puis affectation	<code>\$a = 10 ; \$a *= 5 ; // 50</code>
.	Concaténation	<code>echo 'Ici' . ' et' . ' là' ;</code>
.=	Concaténation et assignation	<code>\$a = 'Ici'; \$a .= ' et'; \$a .= ' là';</code>

Les opérateurs de comparaison

Opérateur	Opération	Exemple
==	Égal à	if (\$i == 1)
<	Inférieur à	if (\$i < 1)
>	Supérieur à	if (\$i > 1)
<=	Inférieur ou égal à	if (\$i <= 1)
>=	Supérieur ou égal à	if (\$i >= 1)
!=	Différent de	if (\$i != 1)
===	Égal à en type et valeur	\$a = 10 ; \$b = '10';
!==	Différent de en type ou en valeur	\$a == \$b ; // true \$a === \$b ; // false

Les opérateurs logiques

Opérateur	Opération	Exemple
!	Négation	!\$a
&&	« Et » logique	if ((\$a == \$b) && (\$c == \$d))
	« Ou » logique	if ((\$a == \$b) (\$c == \$d))
AND	« Et » logique	if ((\$a == \$b) AND (\$c == \$d))
OR	« Ou » logique	if ((\$a == \$b) OR (\$c == \$d))
XOR	« Ou » exclusif	if ((\$a == \$b) XOR (\$c == \$d))

Les conditions

```
<?php
if ($a > $b) {
    echo "a est plus grand que b";
} elseif ($a == $b) {
    echo "a est égal à b";
} else {
    echo "a est plus petit que b";
}
```

Switch

```
switch ($i) {  
    case 0:  
        echo "i égal 0";  
        break;  
    case 1:  
        echo "i égal 1";  
        break;  
    case 2:  
        echo "i égal 2";  
        break;  
}
```

Syntaxe alternative

- Syntaxe utilisée conjointement au HTML (if, while, for, foreach et switch) :

```
<?php if ($a == 5): ?>
```

A égal 5

```
<?php endif; ?>
```

- Syntaxe rapide

```
$action = (empty($_POST['action'])) ? 'default' : $_POST['action'];
```


Les boucles

```
$arr = array(1, 2, 3, 4);  
foreach ($arr as &$amp;value) {  
    $value = $value * 2;  
}
```

```
for ($i = 1; $i <= 10; $i++) {  
    echo $i;  
}
```

```
$i = 1;  
while ($i <= 10):  
    echo $i;  
    $i++;  
Endwhile;
```

```
$i = 0;  
do {  
    echo $i;  
} while ($i > 0);
```

10 – TD n°1

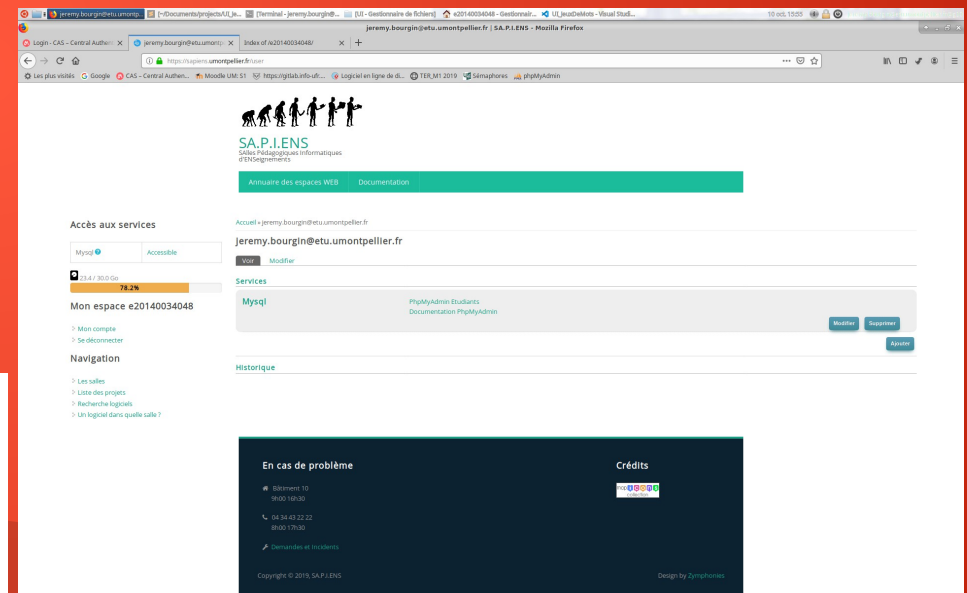
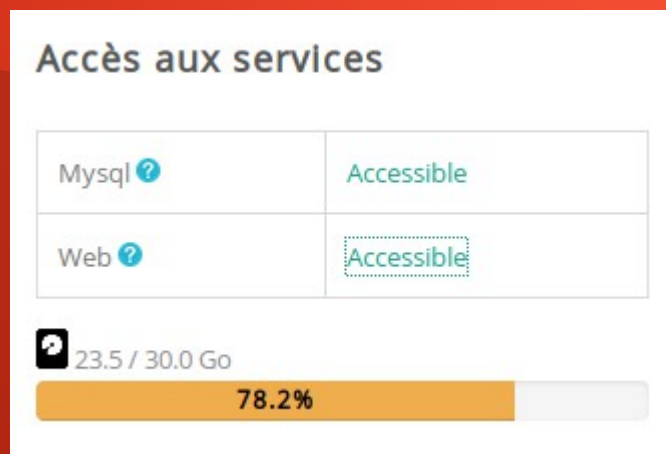
Voir « TD n°1.zip »

- Hello World

TD n°1 : PHP, Hello World

107

- Aller sur **SA.P.I.ENS**.
- Mon compte / Service
- Ajouter les services « Web » & « MySQL ».
- Les services sont accessibles à gauches (URL de votre site web).



- Utiliser la documentation sur l'utilisation du FTP SA.P.IENS.
- Décompressez « TD n°1.zip » dedans votre espace.
- Rendez-vous à l'adresse de votre espace web.
- Si tout ce passe bien vous devriez voir un « Hello world » de trois façons différentes :
 - Procédurale,
 - Fonctionnelle,
 - Objet.

11 – TP n°2

Voir « TP2.tar.gz »

- Acronyme
- Conjecture de Syracuse
- Lecture de fichier

TP n°2 : Les bases de PHP

110

- Décompressez « TP n°2.zip » dans votre racine apache.
- Laissez-vous guider par ce qui s'affiche dans le index.php.

12 – POO

- Classes
- Héritage
- Design Pattern
- PDO

Les classes

```
<?php
class SimpleClass
{
    // déclaration d'une propriété
    public $var = 'une valeur par défaut';

    // Déclaration d'une constante de classe
    const CONSTANT = 'valeur constante';

    // déclaration des méthodes
    public function displayVar() : string {
        echo $this->var;
    }
}
```


Création d'une instance

```
<?php  
$instance = new SimpleClass();  
  
// Ceci peut également être réalisé avec une variable :  
$className = 'Foo';  
$instance = new $className(); // Foo()  
  
?>
```

Héritage

- Comme tout langage objet, PHP permet l'héritage, l'implémentation d'interface, etc...

```
<?php  
class ExtendClass extends SimpleClass  
{  
    // Redéfinition de la méthode parente  
    function displayVar()  
    {  
        echo "Classe étendue\n";  
        parent::displayVar();  
    }  
}
```

```
$extended = new ExtendClass();  
$extended->displayVar();
```

Constructeur et destructeur

- L'approche objet de php permet les `__construct()` et `__destruct()` :

```
<?php
class BaseClass {
    function __construct() {
        print "In BaseClass constructor\n";
    }
}
```

```
class SubClass extends BaseClass {
    function __construct() {
        parent::__construct();
        print "In SubClass constructor\n";
    }
}
```

Typage

- Il est possible de typer la signature d'une fonction ou d'une méthode :

```
<?php
// Un exemple de classe
class MaClasse {
    public $var = 'Bonjour le monde!';
}

/**
 * Fonction de test
 *
 * Le premier paramètre doit être un objet de type MaClasse
 */
function maFonction(MaClasse $foo) : string {
    return $foo->var;
}

$maclasse = new MaClasse;
maFonction($maclasse);
```

Visibilité

- La visibilité d'une propriété, d'une méthode ou (à partir de PHP 7.1.0) une constante peut être définie en préfixant sa déclaration avec un mot-clé : public, protected, ou private. Les éléments déclarés comme publics sont accessibles partout. L'accès aux éléments protégés est limité à la classe elle-même, ainsi qu'aux classes qui en héritent et parente. L'accès aux éléments privés est uniquement réservé à la classe qui les a définis.

```
<?php
```

```
class MaClasse {  
    public $publicVar = 'I'm public';  
    private $privateVar = 'I'm private' ;  
    protected $protectedVar = 'I'm protected' ;  
  
    public function getPrivateVar() : string {  
        return $this->privateVar ;  
    }  
}  
  
$maClass = new MaClasse() ;  
echo $maClass->publicVar ;  
echo $maClass->getPrivateVar() ;
```

Autoloading

- Introduit par PHP 5,
- Permet le chargement dynamique de fichier à partir du nom de la classe.

```
<?php  
function __autoload($class_name) {  
    include classes/$class_name . '.php';  
}
```

```
$obj = new MaClasse1();  
$obj2 = new MaClasse2();
```

```
<?php
class Singleton {
    private static $_instance;

    /**
     * Empêche la création externe d'instances.
     */
    private function __construct () {}

    /**
     * Empêche la copie externe de l'instance.
     */
    private function __clone () {}
```

```
    /**
     * Renvoi de l'instance et initialisation si nécessaire.
     */
    public static function getInstance () {
        if (!(self::$_instance instanceof self))
            self::$_instance = new self();

        return self::$_instance;
    }

    /**
     * Méthodes dites métier
     */
    public function uneAction () {}
}

// Utilisation
Singleton::getInstance()->uneAction();
```

Design pattern : Factory

- La méthode Factory vise donc à instancier des objets selon certains arguments, en dissimulant les détails de cette instantiation.

```
<?php
class MaClasse {
    public static function chargerDriver($driver) {
        if (include_once 'Drivers/' . $driver . '.php') {
            $nomDeClasse = 'Driver_' . $driver;
            return new $nomDeClasse;
        } else {
            throw new Exception ('Driver non trouvé');
        }
    }
}

$mysql = MaClasse::chargerDriver('MySQL');
$sqlite = MaClasse::chargerDriver('SQLite');
```


PDO

- PHP Data Objects est une couche d'abstraction aux bases de données.
- Compatible avec plusieurs SGBD via des drivers : <http://fr2.php.net/manual/en/pdo.drivers.php>

```
<?php
```

```
// Connexion à la BDD
```

```
$connection = new PDO("mysql:host=$host;dbname=$dbName", $dbUser, $dbPwd);
```

```
// On va chercher tous les membres de la table qu'on trie par ordre croissant
```

```
$results = $connection->query("SELECT membre
```

```
    FROM membres
```

```
    ORDER BY membre ASC");
```

```
$resultats->setFetchMode(PDO::FETCH_OBJ);
```

```
// On récupère la liste des membres
```

```
while( $ligne = $resultats->fetch() )
```

```
{
```

```
    echo 'Utilisateur : '.$ligne->membre.'<br />';
```

```
}
```

```
$resultats->closeCursor(); // on ferme le curseur des résultats
```

13 – Normes et PSR

- PSR-0
- PSR-1
- PSR-2
- PSR-3, 4 & next...

Introduction

- PHP est un langage anarchique car il utilise plusieurs conventions :
 - snake_case: `file_get_contents()`,
 - Tout attaché : `strtolower()`,
 - lowerCamelCase : `DateTime::setISODate()`.
- PHP est permissif,
- Bref, nous avons besoin de norme !
- Les PSR permettent l'interopérabilité entre les framework.
- Plus d'infos : <http://www.php-fig.org/>

PSR-1 (La norme de codage de base)

- Les fichiers DOIVENT utiliser seulement les tags `<?php` et `<?=`.
- Les fichiers de code PHP DOIVENT être encodés uniquement en UTF-8 sans BOM.
- Les fichiers DEVRAIENT soit déclarer des symboles (classes, fonctions, constantes, etc.) soit causer des effets secondaires (par exemple, générer des sorties, modifier paramètres .ini), mais NE DOIVENT PAS faire les deux.
- Les espaces de noms et les classes DOIVENT suivre PSR-0.
- Les noms des classes DOIVENT être déclarés comme StudlyCaps.
- Les constantes de classe DOIVENT être déclarées en majuscules avec un sous-tiret en séparateurs.
- Les noms des méthodes DOIVENT être déclarés comme camelCase.

<http://www.php-fig.org/psr/psr-1/>

PSR-2 (Guide pour le style d'écriture de code)

- Le code DOIT suivre les PSR-1.
- Le code DOIT utiliser 4 espaces pour l'indentation et aucune tabulation.
- Il NE DOIT PAS exister une limite stricte sur la longueur de la ligne, la limite acceptable DOIT être de 120 caractères; les lignes DEVRAIENT comprendre 80 caractères ou moins.
- Il DOIT y avoir une ligne vide après la déclaration de l'espace de noms, et il DOIT y avoir une ligne vide après le bloc de déclarations use.
- L'ouverture des accolades pour les classes DOIT figurer sur la ligne suivante, les accolades de fermeture DOIVENT figurer sur la ligne suivante après le corps de la classe.
- L'ouverture des accolades pour les méthodes DOIT figurer sur la ligne suivante, les accolades de fermeture DOIVENT figurer sur la ligne suivante après le corps de la méthode.
- La visibilité DOIT être déclarée sur toutes les propriétés et méthodes; abstraite et finale doivent être déclarés avant la visibilité; statique DOIT être déclaré après la visibilité.

<http://www.php-fig.org/psr/psr-2/>

PSR-2 (Guide pour le style d'écriture de code)

- La structure des mots-clés de contrôle DOIT avoir un espace après eux, les méthodes et les appels de fonction NE DOIVENT PAS en avoir.
- L'ouverture des accolades pour les structures de contrôle DOIT figurer sur la même ligne, et la fermeture des accolades DOIT figurer sur la ligne suivante après le corps.
- l'ouverture des parenthèses pour les structures de contrôle NE DOIT PAS contenir d'espace après eux, la fermeture de parenthèses pour les structures de contrôle NE DOIT PAS contenir d'espace avant.

PSR-3, 4, ... mais aussi les draft...

Num	Title	Editor	Coordinator	Sponsor
1		Paul M. Jones	N/A	N/A
2		Paul M. Jones	N/A	N/A
3		Jordi Boggiano	N/A	N/A
4		Paul M. Jones	Phil Sturgeon	Larry Garfield
6		Larry Garfield	Paul Dragoonis	Robert Hafner
7		Matthew Weier O'Phinney	Beau Simensen	Paul M. Jones
11		Matthieu Napoli, David Négrier	Matthew Weier O'Phinney	Korvin Szanto
13		Larry Garfield	Matthew Weier O'Phinney	Marc Alexander
16		Paul Dragoonis	Jordi Boggiano	Fabien Potencier

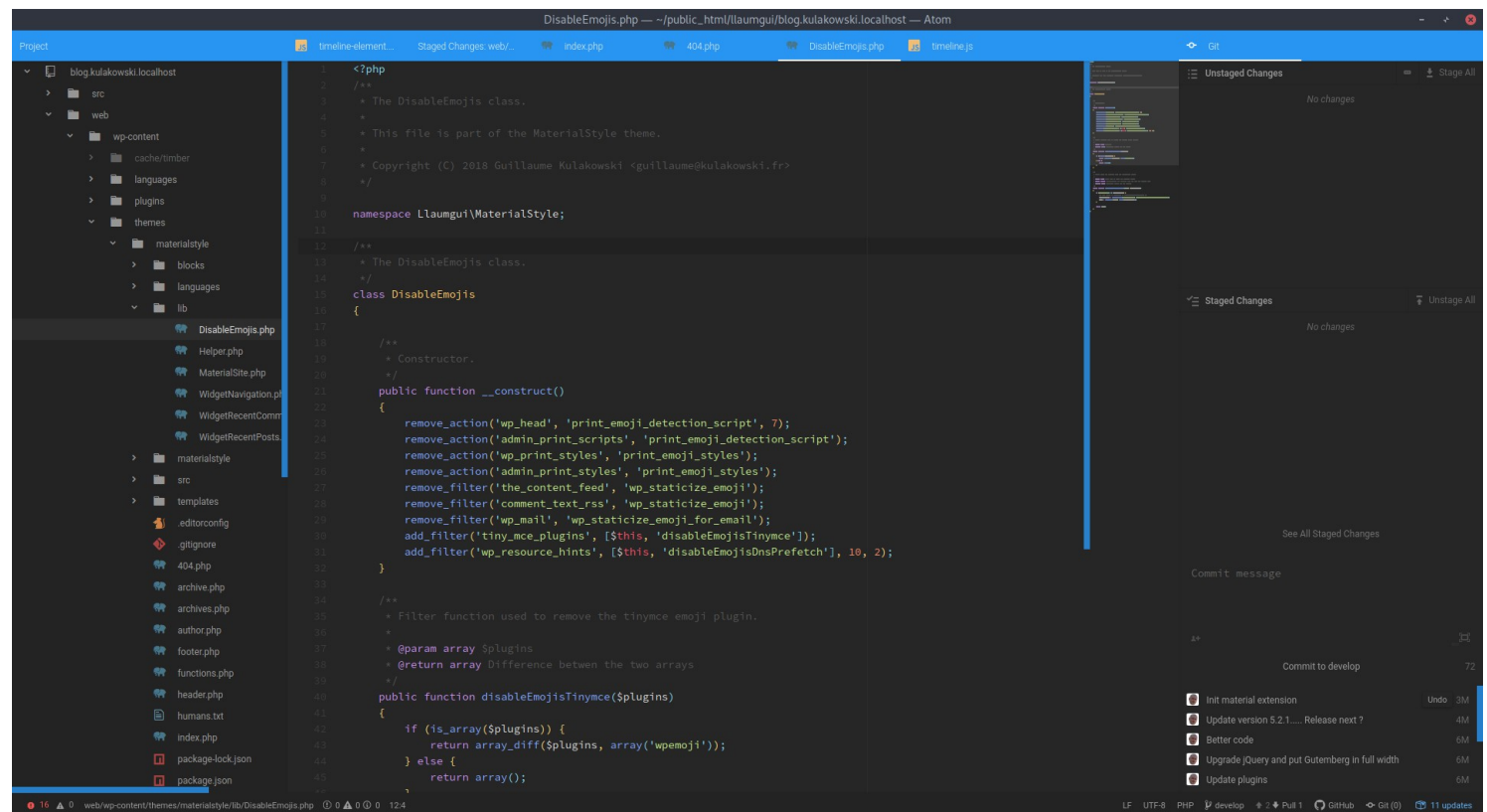
14 – Les outils

- Netbeans
- Eclipse
- PHPMyAdmin
- MySQL Work Bench

EDI

Bien entendu il existe moult éditeurs de texte / EDI :

- Atom
- Brackets
- Eclipse
- Netbeans
- Notepad++
- PhpStorm
- Sublime Text
- Etc...



PHPMyAdmin

- WebUI pour manipuler des bases de données MySQL / MariaDB,
- Open Source,
- Intégré à WAMP.

The screenshot displays the PHPMyAdmin interface for the 'wordpress' database, specifically the 'wp_comments' table. The table structure is shown with columns like 'comment_ID', 'comment_post_ID', 'comment_author', etc. Below the structure, there are sections for 'Index' and 'Statistiques' (Statistics).

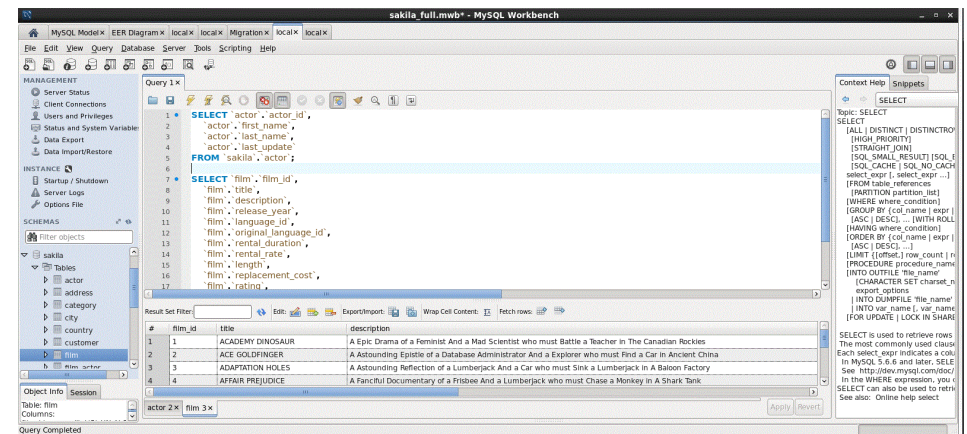
Index				Espace utilisé		Statistiques	
Nom de l'index	Type	Cardinalité	Action	Type	Espace	Information	Valeur
PRIMARY	PRIMARY	371		Données	820,7 Kio	format	dynamique
comment_approved	INDEX	aucune		Index	256,8 Kio	Interclassement	utf8_general_ci
comment_post_ID	INDEX	aucune		Perte	440,5 Kio	Enregistrements	371
comment_approved_date_gmt	INDEX	aucune		effectif	696,2 Kio	Longueur enr. ø	1 949
comment_date_gmt	INDEX	aucune		Total	1 076,7 Kio	Taille enr. ø	2 972 ø

Créer un index sur 1 colonne(s) Exécuter

Statistiques: Ven 03 Octobre 2009 à 20:08
Création
Dernière modification: Mer 07 Janvier 2009 à 05:59

MySQL Work Bench

- Le client lourd (dans tous les sens du terme) pour manipuler des bases de données MySQL / MariaDB,
- Open Source.



15 – TP n°3

Voir « TP n°3.tar.gz »

- Classe
- POO
- PDO

Annexes

TP n°3 : Plus loin avec PHP...

134

- Décompressez « TP n°3.zip » dans votre racine apache.
- Laissez-vous guider par ce qui s'affiche dans le index.php.
- Concernant le db_config.php, vous allez devoir vous créer un compte sur *SA.P.IENS* pour avoir à disposition une base MySQL.

Webographie

- Le site & les forums d'AlsaCreations : <http://www.alsacreations.com/>
- Le site du W3C : <http://www.w3.org/>
- Les bonnes pratiques Yahoo ! : <http://developer.yahoo.com/performance/>
- API jQuery : <http://api.jquery.com/>
- Centre d'apprentissage jQuery : <http://learn.jquery.com/>
- Site officiel de PHP : <http://www.php.net/>
- Le manuel PHP : <http://php.net/manual/fr/index.php>
- OpenClassRooms : <https://openclassrooms.com/>

Bibliographie

- HTML5, une référence pour le développeur Web.
Rodolphe Rimelé.
(ISBN : 978-2-212-12982-3).
- CSS avancées, vers HTML5 et CSS3.
Raphaël Goetter.
(ISBN : 978-2-212-13405-6).
- PHP 7 avancé.
Eric Daspet, Cyril Pierre de Geyer Julien Pauli & Pascal Martin.
(ISBN : 978-2-212-143577).