

Formation HTML/CSS

Romain Bohdanowicz

Twitter : @bioub

Github : <https://github.com/bioub>

Présentations

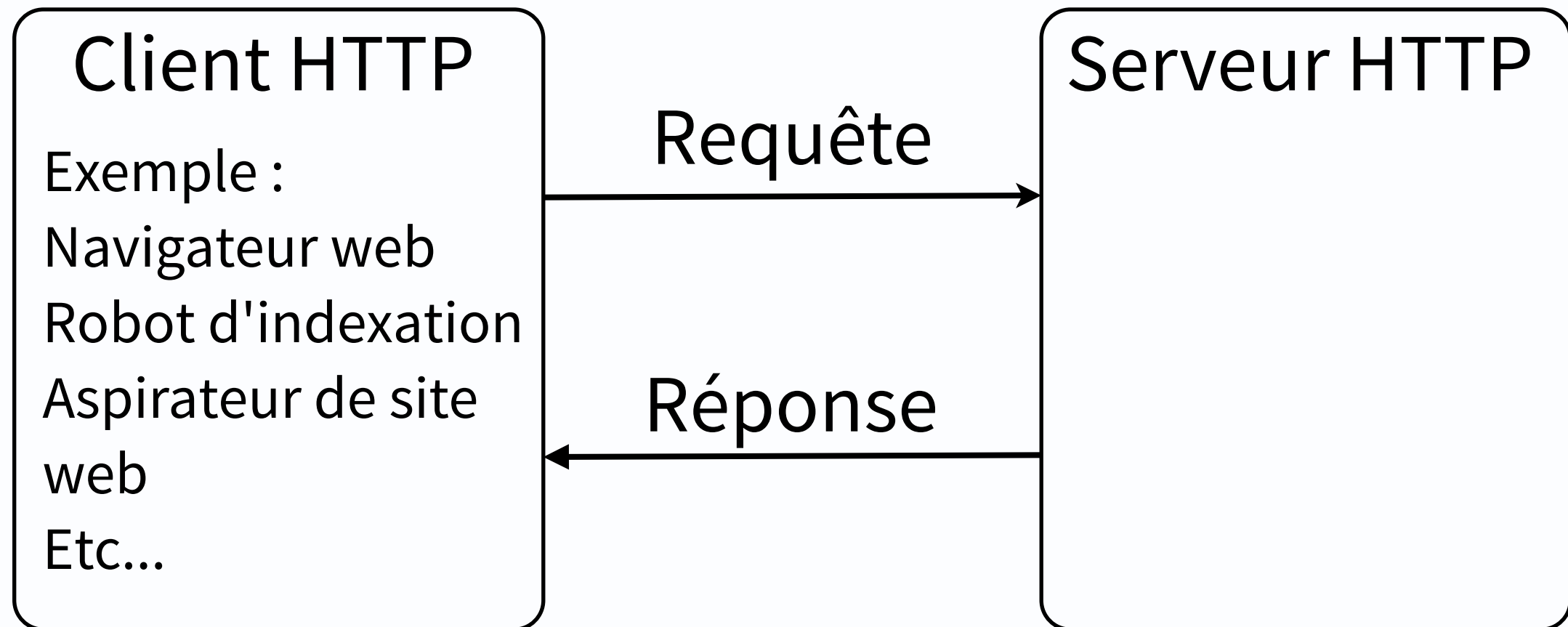
- Romain Bohdanowicz
Ingénieur EFREI 2008, spécialité en Ingénierie Logicielle
- Expérience
Formateur/Développeur Freelance depuis 2006
Plus de 8000 heures de formation animées
- Langages
Expert : HTML / CSS / JavaScript / PHP / Java
Notions : C / C++ / Objective-C / C# / Python / Bash / Batch
- Certifications
PHP 5 / PHP 5.3 / PHP 5.5 / Zend Framework 1
- Particularités
Premier site web à 12 ans (HTML/JS/PHP), Triathlète à mes heures perdues
- Et vous ?
Langages ? Expérience ? Utilité de cette formation ?

HTML - Introduction

Introduction

- Fin des années 80 : Tim Berners-Lee, chercheur au CERN créé le web :
 - HTML
 - URL
 - Protocole HTTP
 - Navigateur Web
 - Serveur Web

Protocole HTTP



Protocole HTTP

- Requête HTTP

```
MacBook-Pro-de-Romain:~ romain$ telnet www.google.fr 80
Trying 216.58.213.3...
Connected to www.google.fr.
Escape character is '^]'.
GET / HTTP/1.1
User-Agent: Romain depuis Telnet
Host: www.google.fr
```

Protocole HTTP

▸ Réponse HTTP

HTTP/1.1 200 OK

Date: Fri, 08 Apr 2016 09:23:51 GMT

Expires: -1

Cache-Control: private, max-age=0

Content-Type: text/html; charset=ISO-8859-1

Server: gws

Transfer-Encoding: chunked

4b7d

```
<!doctype html><html itemscope="" itemtype="http://schema.org/  
WebPage" lang="fr"><head><meta content="text/html; charset=UTF-8"  
http-equiv="Content-Type"><meta content="/images/branding/googleg/  
1x/googleg_standard_color_128dp.png"  
itemprop="image"><title>Google</title>
```

...

Langages du Navigateur

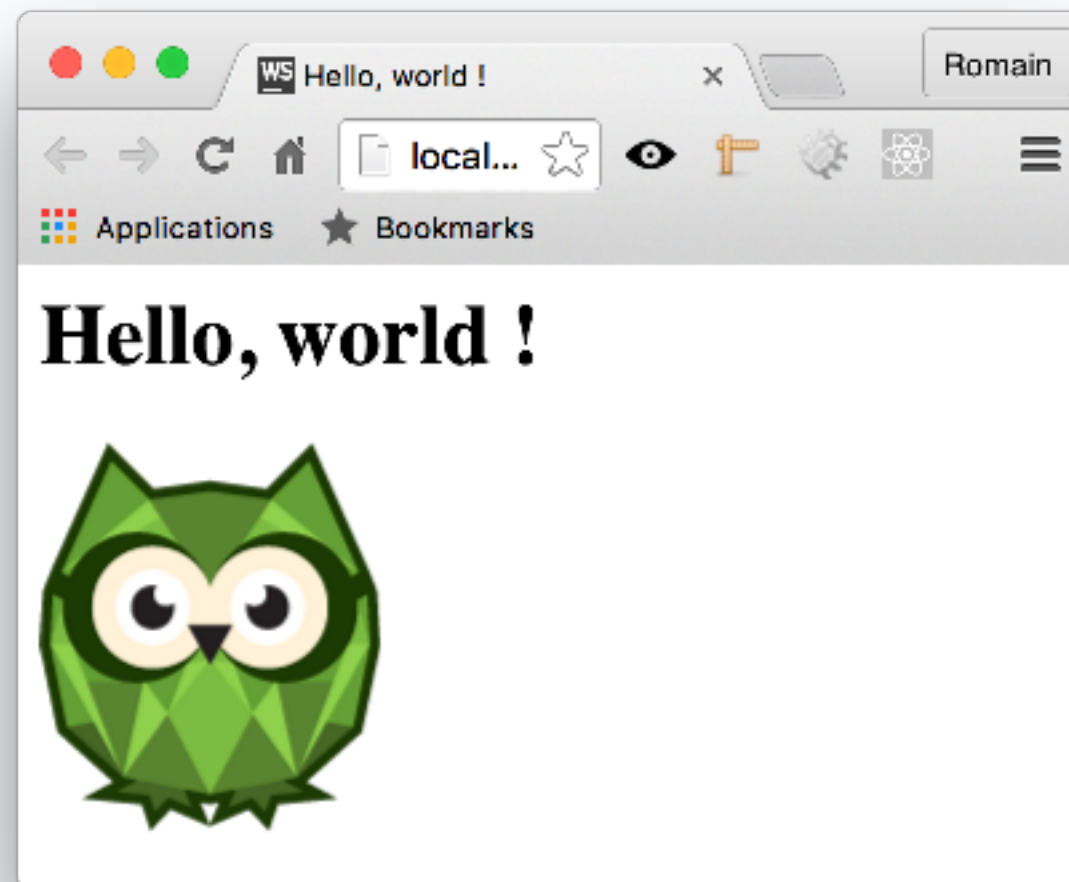
- Principaux langages compris par les navigateurs modernes
 - HTML : structure de la page (apprentissage facile)
 - CSS : mise en forme (apprentissage intermédiaire)
 - JavaScript : interactions dynamiques avec l'utilisateur ou le serveur (apprentissage difficile)
- D'autres moins connus :
 - SVG, RSS, ATOM, MathML, ...
- Langages du serveur :
 - PHP, Java, C#, Ruby, Python, Go, JavaScript...

Normes HTML

- Différentes époques :
 - Jusqu'à HTML 3.2, pas de norme
 - HTML 3.2 à HTML 4.01, norme du W3C basé sur SGML
 - XHTML 1.0 et 1.1, norme du W3C basé sur XML
 - HTML5 ou HTML Living Standard, norme du WHATWG, reprise par le W3C

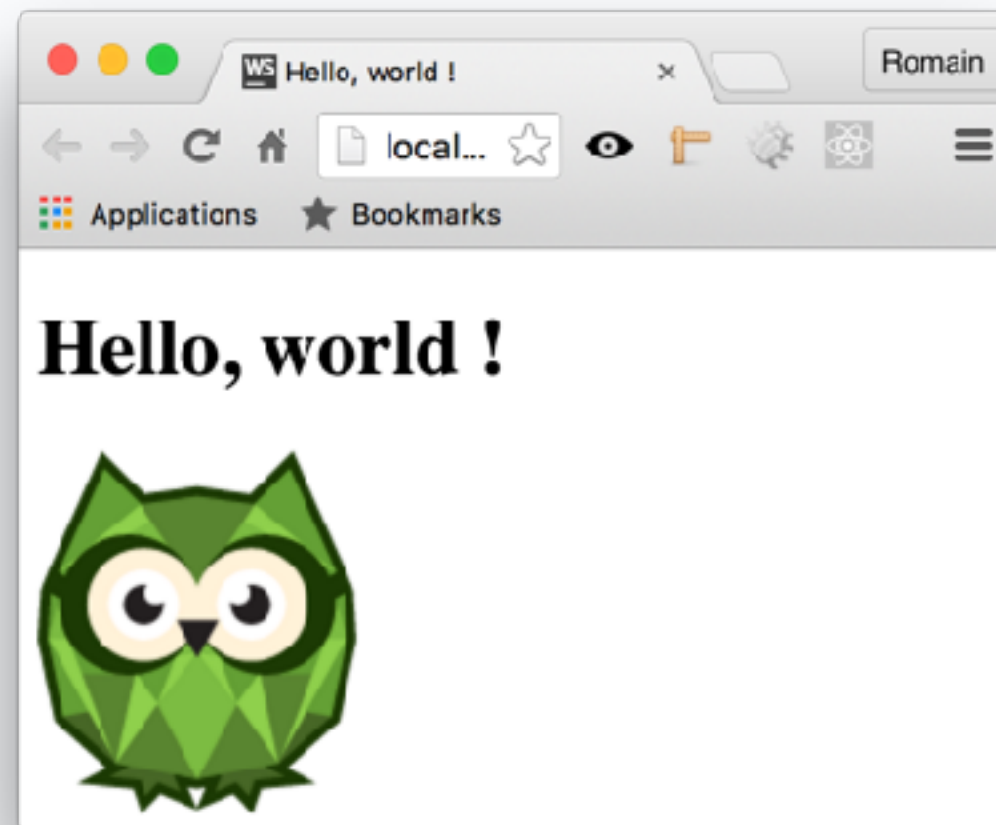
HTML sans norme

```
<HTML>
<HEAD>
  <TITLE>Hello, world !</TITLE>
</HEAD>
<BODY BGCOLOR=#FFFFFF>
  <H1>Hello, world !</H1>
  <IMG SRC=logo-formation.tech.png>
</BODY>
</HTML>
```



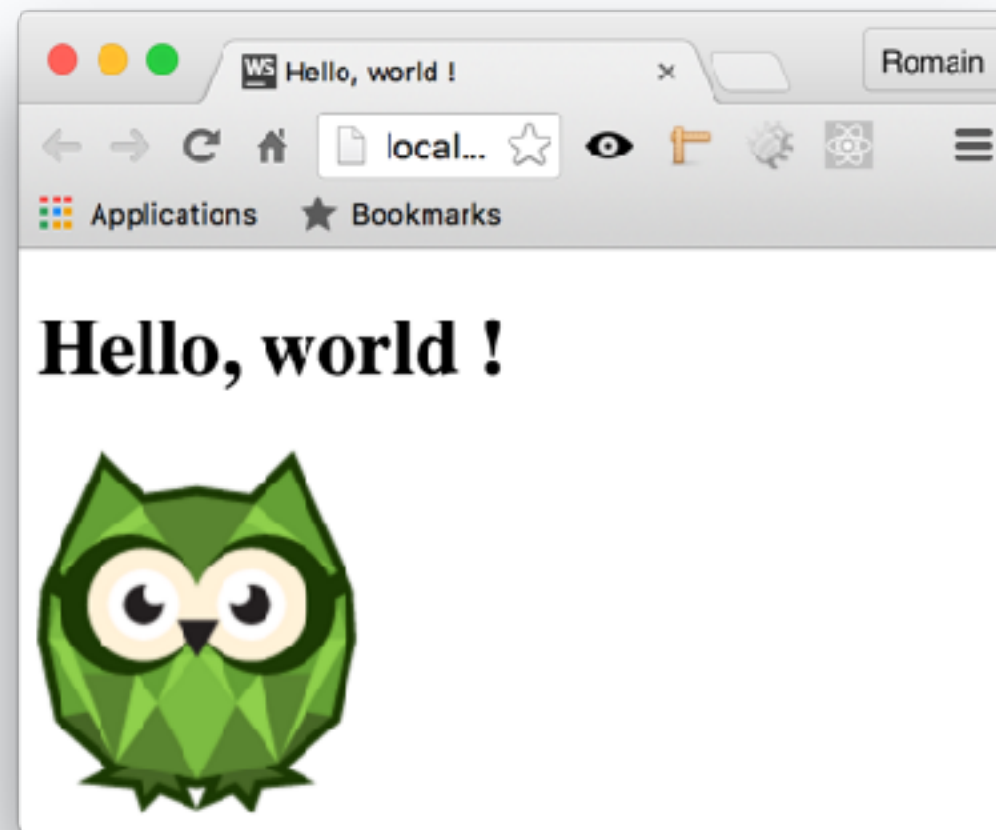
HTML 4.01

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
  <title>Hello, world !</title>
</head>
<body>
  <h1>Hello, world !</h1>
  <img src=logo-formation.tech.png alt="Logo Formation.tech">
</body>
</html>
```



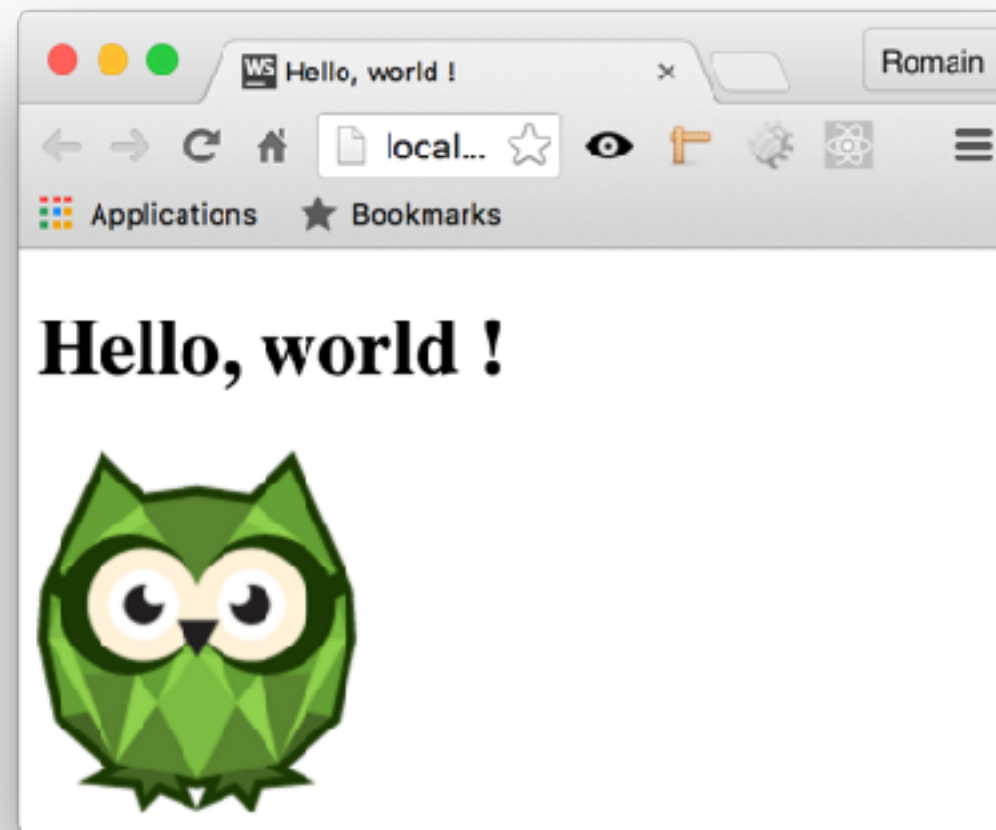
XHTML 1.0

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html
    PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>Hello, world !</title>
</head>
<body>
    <h1>Hello, world !</h1>
    
</body>
</html>
```



HTML5

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Hello, world !</title>
</head>
<body>
  <h1>Hello, world !</h1>
  
</body>
</html>
```



A propos d'HTML5

▸ HTML5

- Méésentente entre le W3C et les éditeurs de navigateurs qui finissent par créé leur propre organisme de normalisation : le WHATWG
- Normer l'existant
- Demander aux utilisateurs du langages de
- Normer la correction d'erreur

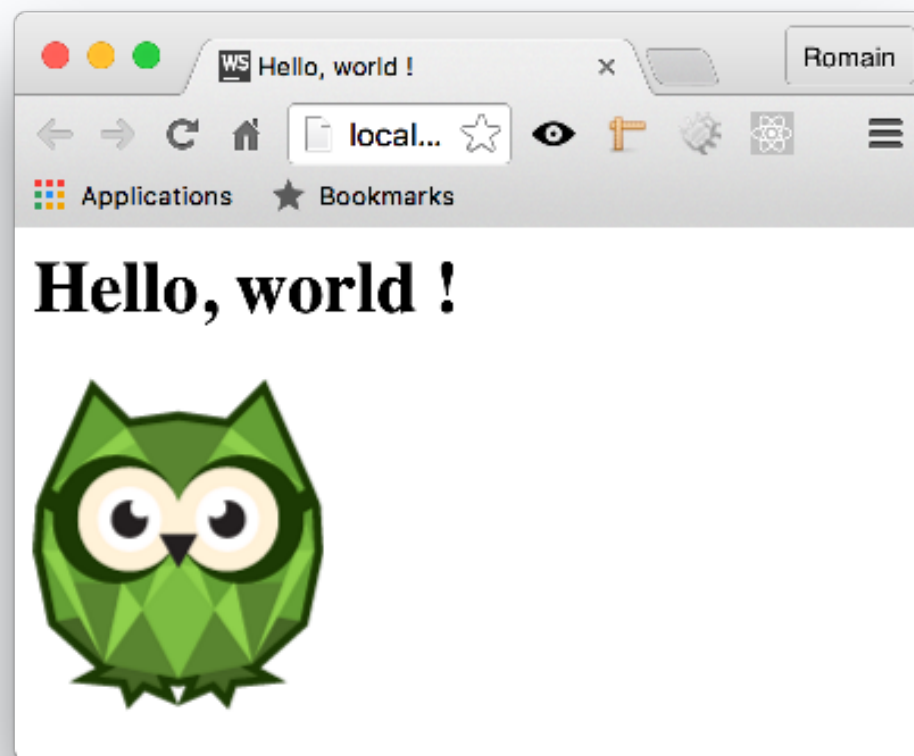
▸ HTML5, un terme marketing

- Le successeur de Web 2.0
- Nouveautés HTML
- Nouveautés CSS (principalement CSS3)
- Nouveaux APIs du Navigateur (Géolocalisation, Notifications, ...)

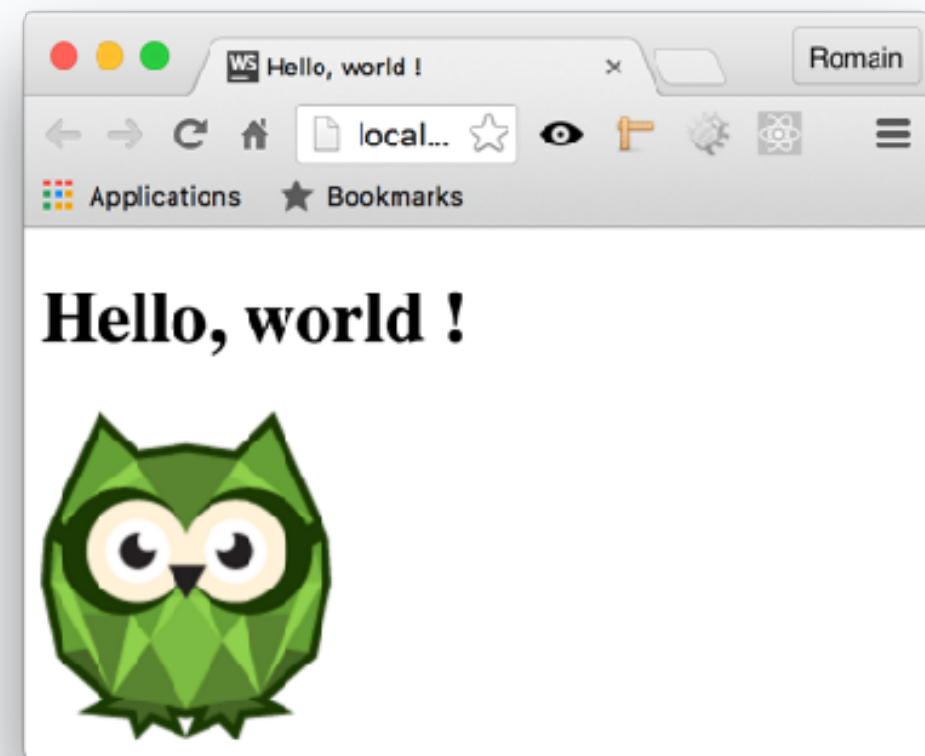
Différences de rendu

- Différences de rendu
 - Le calcul des marges est différent
 - N'oubliez pas le DOCTYPE !

HTML non normé

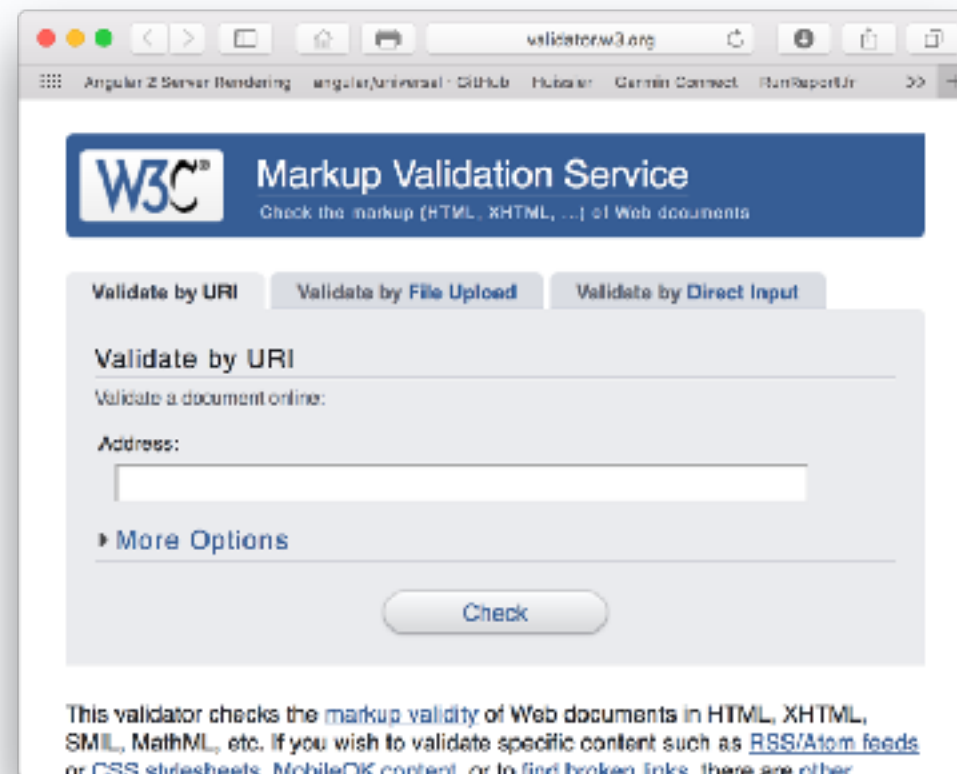


HTML normé



Valider son HTML

- ▶ Vérifier que son Navigateur respecte bien la norme annoncée :
 - ▶ <http://validator.w3.org>
- ▶ Pour être valide :
 - ▶ Respecter la syntaxe (<, >, ", ...)
 - ▶ Respecter la grammaire (mots clés, structure...)



Exercices

- ▶ Exercices
 - ▶ Aller sur le site de votre choix, afficher la source. Quelle est la norme utilisée ?
 - ▶ Valider ce site sur le validateur du W3C

HTML - Structure d'une page web

Balise

- Un fichier HTML est une succession de « balises », imbriquées les unes dans les autres
- Une balise peut être ouvrante, fermante, ou vide (pas de contenu), il faut connaître le langage pour savoir celle qui sont vide ou non.
- Format d'une balise avec une partie ouvrante puis fermante :
 - **<nomDeLaBalise></nomDeLaBalise>**
- Format d'une balise vide :
 - **<baliseVide>**

Imbrication

- Les balises d'une page web sont imbriquées les unes dans les autres

- Exemple :

```
<baliseParent>  
  <baliseEnfant>Contenu</baliseEnfant>  
</baliseParent>
```

- Pour faciliter la lecture du HTML, à chaque fois que l'on imbrique une balise dans une autre, on décale la balise enfant de 4 espaces (parfois 2) : on appelle cela l'indentation.
- Les balises ne peuvent pas se chevaucher (penser à des dossiers dans des dossiers) :

```
<baliseParent>  
  <baliseEnfant></baliseParent>  
</baliseEnfant>
```

Attributs

- Parfois les balises contiennent des informations secondaires qu'on ne souhaite pas afficher à l'écran

- Exemple :

```
<balise attribut="valeur" autre="hello">Contenu à afficher</balise>
```

Structure minimale

- A minima une page HTML contient les balises suivantes :

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
</body>
</html>
```

- DOCTYPE pour indiquer la norme utilisées (attention sans DOCTYPE le rendu est différent)
- html : balise racine qui englobe obligatoirement toutes les autres
- head : toutes les données secondaires qu'on affiche pas sur la pages, ou indirectement (chargement de fichiers CSS, titre de l'onglet...)
- body : contenu de la page

Exercices

- ▶ Exercices
 - Créer un nouveau fichier HTML : index.html
 - Remplir ce fichier avec la structure minimale d'une page web

HTML - Entêtes d'une page web (balise head)

Balise head

- Entête de la page

Le contenu de la balise head n'a pas vocation à s'afficher sur la page, mais est plutôt utilisé par les programmes qui vont lire le HTML (navigateurs, moteurs de recherche, mobiles...)

Encodage

- L'encodage de la page

Un ordinateur ne manipule que des nombres (computer en anglais, « machine qui compte »), les caractères n'existent pas réellement, on leur a donc attribué un code sous la forme d'un nombre, la lettre A majuscule a par exemple le code 65 quand le e minuscule le code 101.

- Différents encodages

Aujourd'hui l'encodage le plus universel est l'UTF-8, mais il en existe d'autres : windows-latin-1 ou iso-8859-1, ce qui explique que parfois les lettres accentuées n'apparaissent pas correctement lorsque l'encodage est mal renseigné.

- Spécifier l'encodage d'un page web dans les 512 premiers caractères de la page avec :

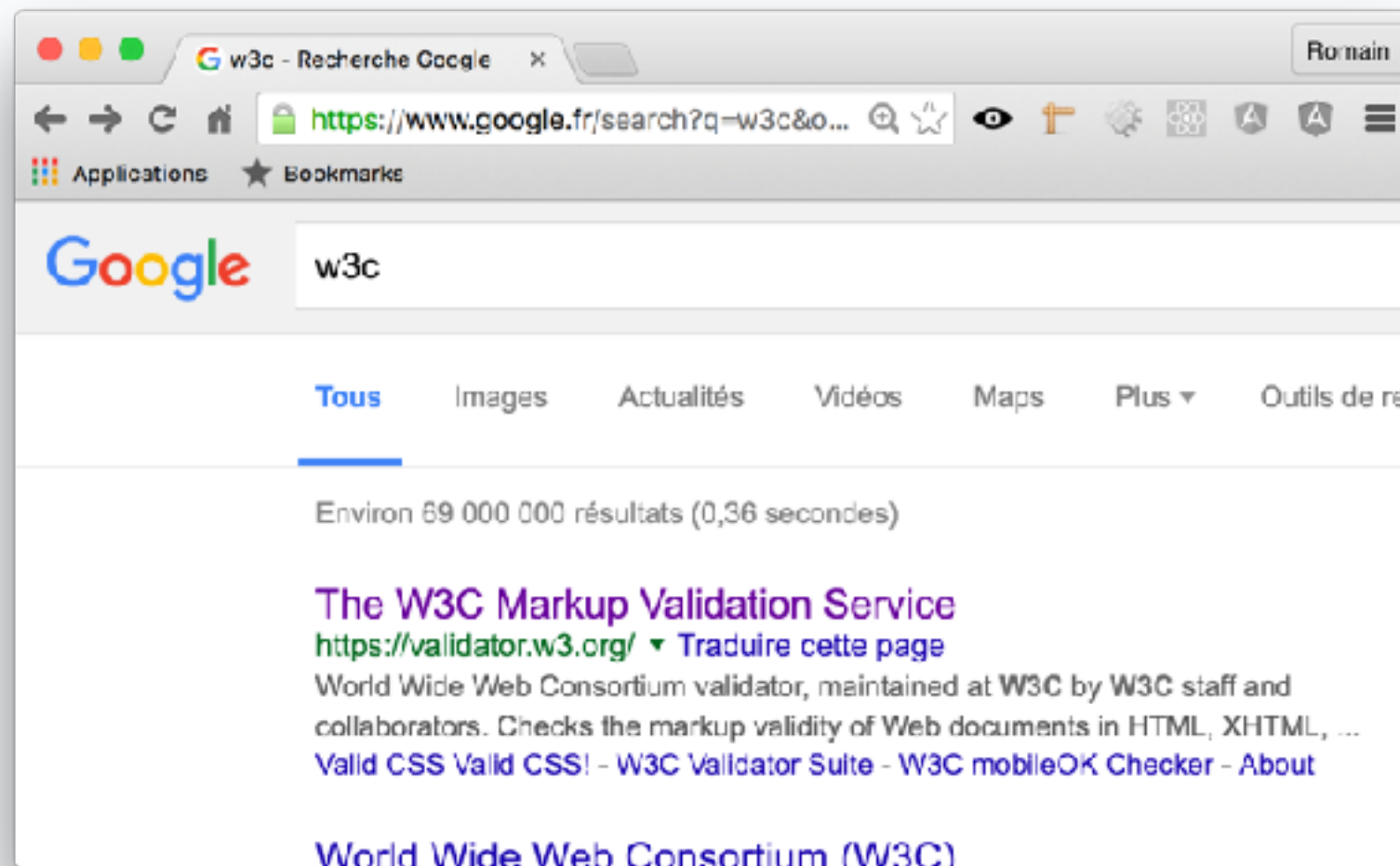
```
<meta charset="UTF-8">
```

```
<meta charset="ISO-8859-1">
```

Titre de la page

- Titre de la page
 - S'affichera dans l'onglet du navigateur
 - Titre utilisé par les moteurs de recherche ou autres programmes

<title>The W3C Markup Validation Service**</title>**



Moteurs de recherche

▸ Utile pour les moteurs de recherche :

- Mots clés (de moins en moins utilisés)

```
<meta name="keywords" content="HTML, HyperText Markup Language, Validation,  
W3C Markup Validation Service">
```

- Description

```
<meta name="description" content="W3C's easy-to-use  
markup validation service, based on SGML and XML parsers.">
```

- Robots (noindex : ne pas indexer la page, nofollow : ne pas suivre les liens présents dans la page)

```
<meta name="robots" content="noindex, nofollow">
```

CSS

- Dans la balise head on peut inclure un CSS présent dans un fichier externe avec la balise link

```
<link rel="stylesheet" href="fichier.css">
```

- Ou bien directement dans la balise style

```
<style>  
  body {  
    background: beige;  
  }  
</style>
```

JavaScript

- Dans la balise head on peut inclure un JavaScript présent dans un fichier externe (ne pas oublier la balise `</script>` fermante)

```
<script src="fichier.js"></script>
```

- Ou bien directement dans la balise script

```
<script>
    window.addEventListener('load', function() {
        console.log('Chargement terminé');
    });
</script>
```

Exercices

- ▶ Exercices
 - Créer un nouveau fichier style.css et le charger dans la balise HEAD
 - Créer un nouveau fichier script.js et le charger dans la balise HEAD
 - Ajouter un titre et une balise META description pour le référencement

HTML - Contenu d'une page web (balise body)

Commentaires HTML

► Pour commenter le HTML

- Permet de décrire les balises pour un meilleure relecture par les développeurs
- Permet de désactiver les balises, sans les supprimer

► Exemple

```
<!-- Le menu principal du site -->  
<nav id="menu">  
  <ul>  
    <li><a href="index.html">Accueil</a></li>  
    <li><a href="presentation.html">Présentation</a></li>  
    <li><a href="contact.html">Contact</a></li>  
  </ul>  
</nav>  
<!-- <p>Un paragraphe désactivé</p> -->
```

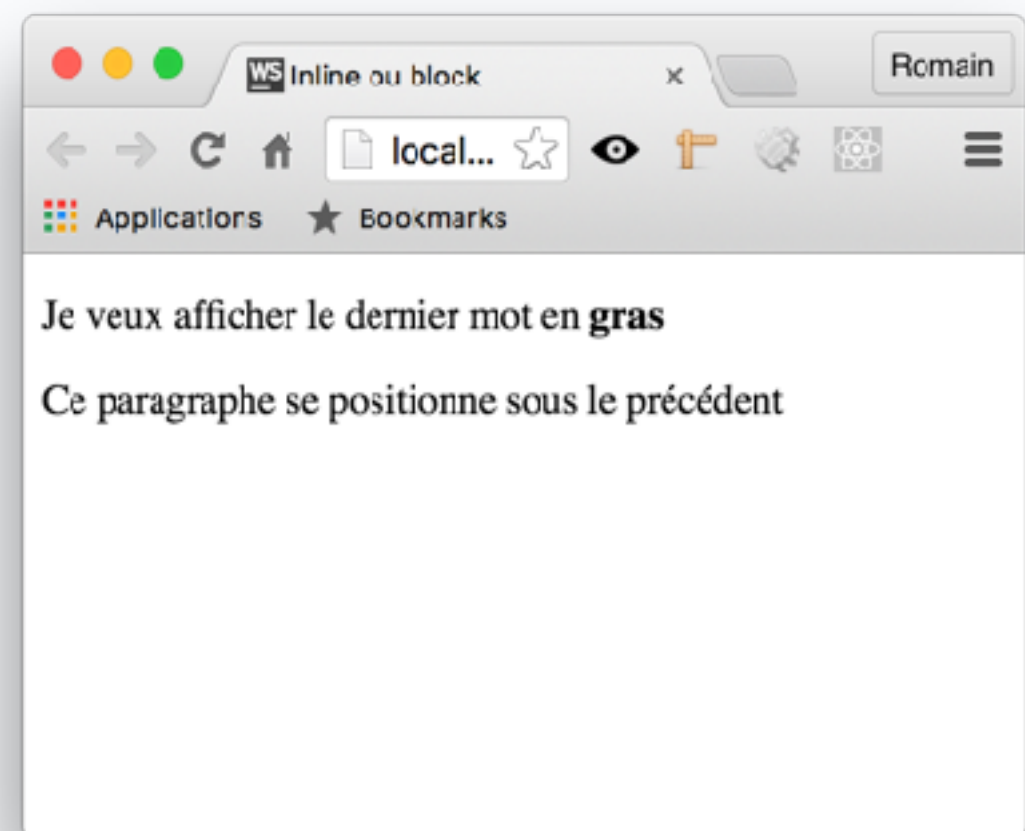
Balises block ou inline

- ▶ En HTML il existe 2 grands types de balises :
 - Balises blocs (block), qui provoquent par défaut un retour à la ligne
 - Balises en ligne/à la file (inline), qui permettent se positionnent au sein d'un texte

- ▶ Exemple

```
<p>
    Je veux afficher le
dernier mot en <b>gras</b>
</p>

<p>
    Ce paragraphe se
positionne sous le précédent
</p>
```



Balises block

- ▶ Principales balises block :

<address>, <article>, <aside>, <blockquote>, <canvas>, <dd>, <div>, <dl>, <fieldset>, <figcaption>, <figure>, <footer>, <form>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <header>, <hgroup>, <hr>, , <main>, <nav>, <noscript>, , <output>, <p>, <pre>, <section>, <table>, <tfoot>, , <video>

- ▶ Principales balises inline :

, <big>, <i>, <small>, <tt>, <abbr>, <acronym>, <cite>, <code>, <dfn>, , <kbd>, , <samp>, <time>, <var>, <a>, <bdo>,
, , <map>, <object>, <q>, <script>, , <sub>, <sup>, <button>, <input>, <label>, <select>, <textarea>

Les balises de regroupement

- Pour faire des regroupements sur une page :
 - `<address>` : coordonnées de contact
 - `<article>` : un article (de blog par exemple)
 - `<aside>` : du contenu associé au contenu principal (colonne de droite, encart au milieu d'un article de presse)
 - `<div>` : regroupement pour la mise en forme (couleur de fond, positionnement)
 - `<header>` : entête (de page, d'article, d'une section quelconque)
 - `<main>` : partie principale (de page, d'article, d'une section quelconque)
 - `<footer>` : pied (de page, d'article, d'une section quelconque)
 - `<nav>` : éléments de navigation
 - `<section>` : regroupement par catégorie (exemple : introduction, présentation...)
- En dehors de div et address, ces balises sont apparues en HTML5, pour certains navigateurs anciens il faut inclure un script de compatibilité : html5shiv

Autres balises bloc courantes

- Les titres : h1, h2, h3, h4, h5, h6

`<h1>Titre 1</h1>`

`<h2>Titre 2</h2>`

`<h3>Titre 3</h3>`

`<h4>Titre 4</h4>`

`<h5>Titre 5</h5>`

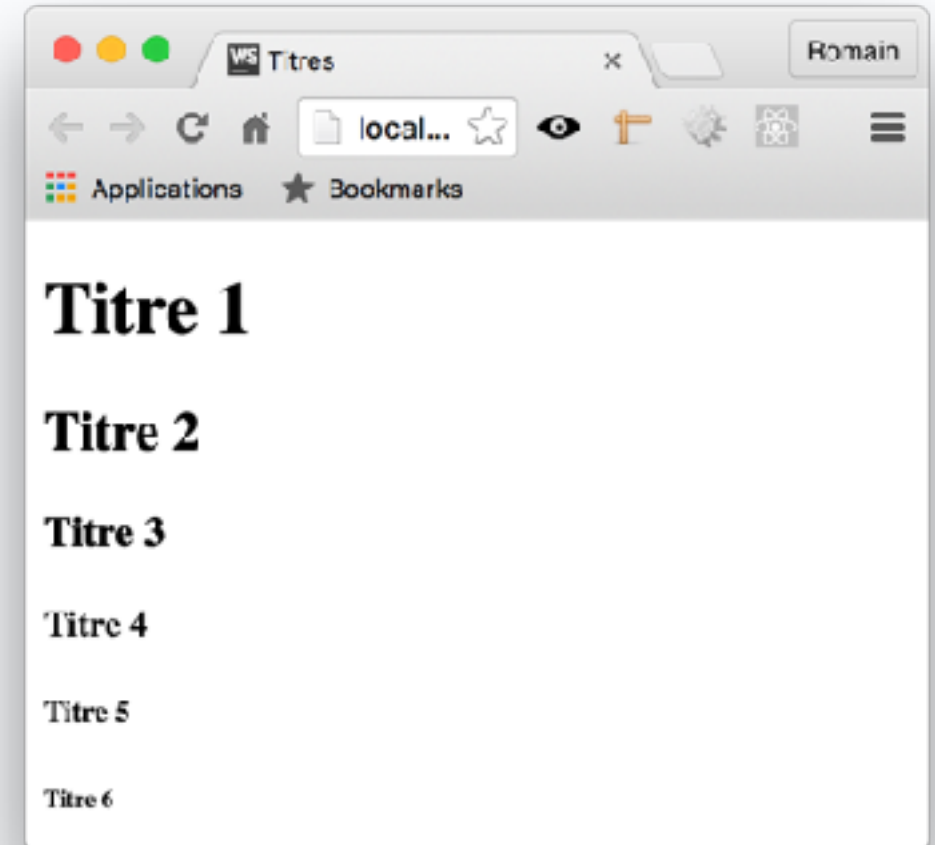
`<h6>Titre 6</h6>`

- Les paragraphes : p

`<p>Un paragraphe</p>`

- Un bloc de citation

`<blockquote>Une citation</blockquote>`



Les listes

▸ Liste non ordonnées

```
<ul>
  <li>Google</li>
  <li>Microsoft</li>
  <li>Apple</li>
</ul>
```

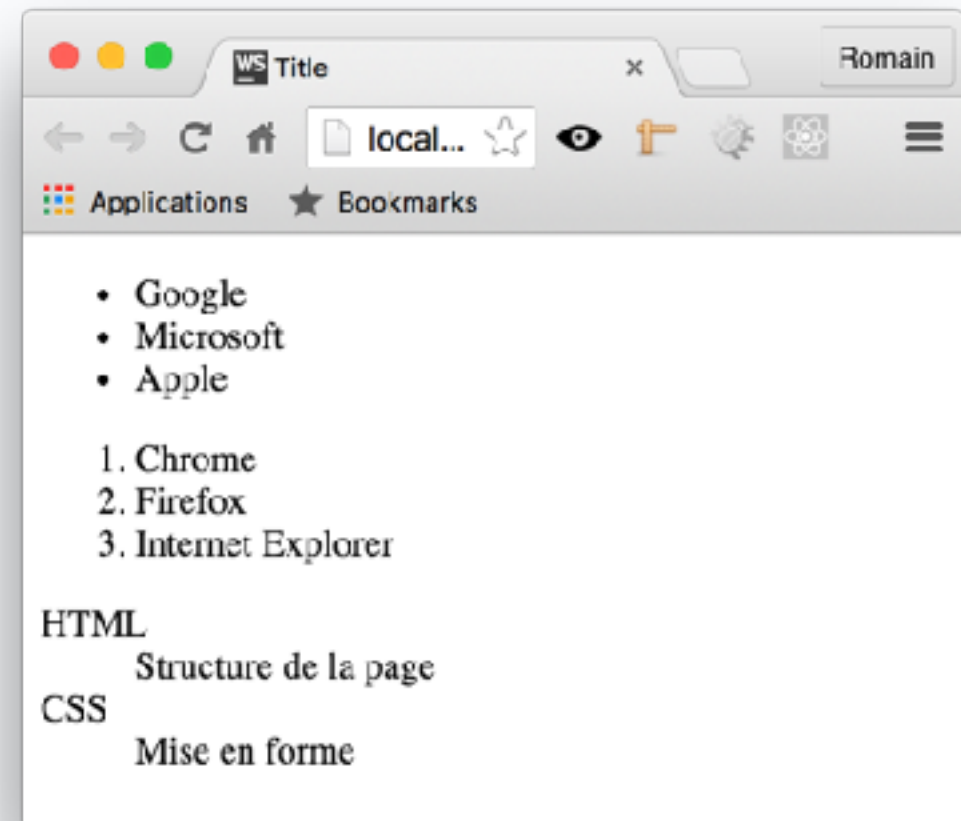
▸ Liste ordonnées

```
<ol>
  <li>Chrome</li>
  <li>Firefox</li>
  <li>Internet Explorer</li>
</ol>
```

▸ Liste de définitions

```
<dl>
  <dt>HTML</dt>
  <dd>Structure de la page</dd>

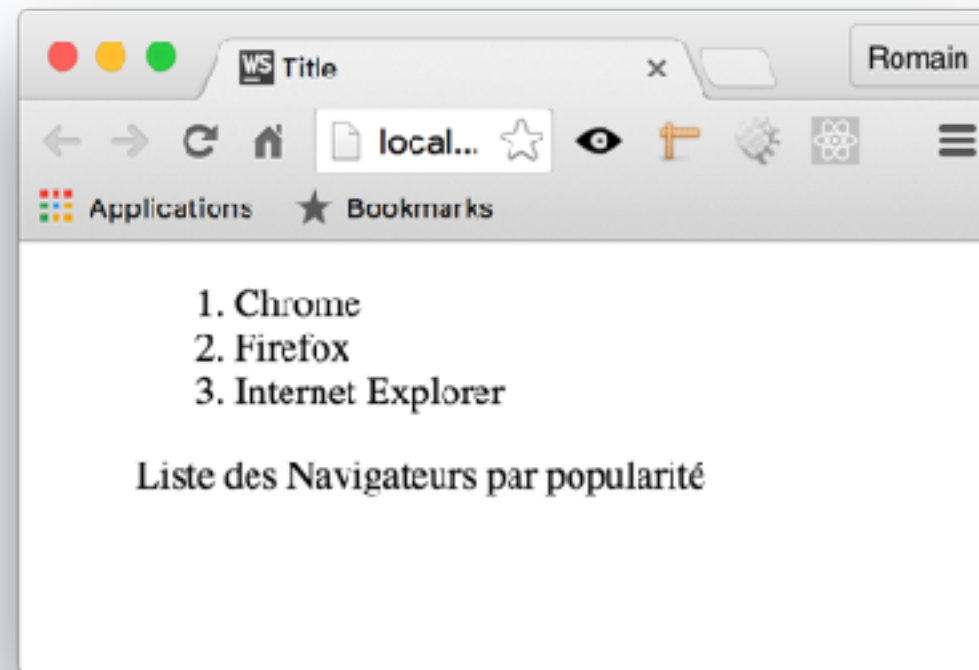
  <dt>CSS</dt>
  <dd>Mise en forme</dd>
</dl>
```



Légende

- Mettre une légende à un élément (liste, tableau, image, vidéo...)

```
<figure>
  <ol>
    <li>Chrome</li>
    <li>Firefox</li>
    <li>Internet Explorer</li>
  </ol>
  <figcaption>Liste des Navigateurs par popularité</figcaption>
</figure>
```



Principales balises inline

► Balises inline

Ne provoque pas de retour à la ligne. Permettent de délimiter des mots.

b : gras, texte sans importance particulière

i : italique, ton différent, parole prise par un autre

em : emphase légère (importance plus grande que le reste du texte)

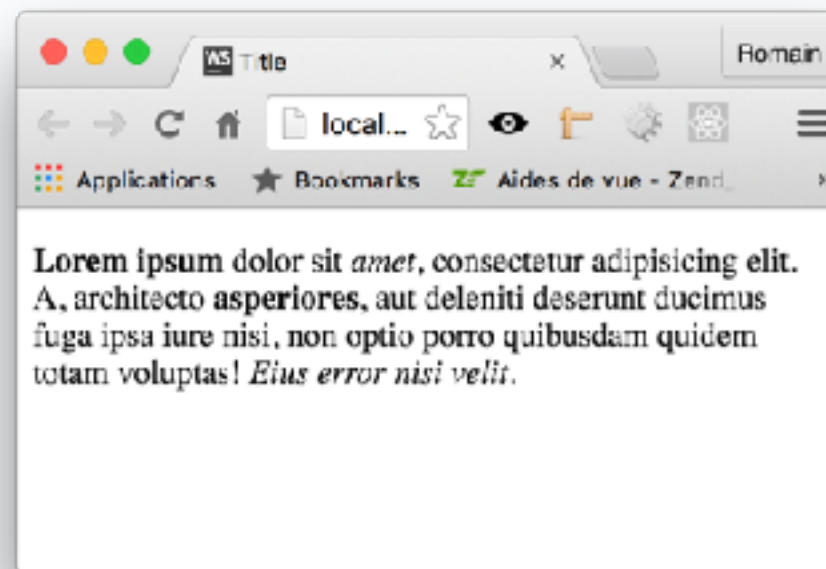
strong : emphase forte

span : délimitation neutre (pour le CSS ou le JS)

`<p>`

```
<b>Lorem ipsum</b> dolor sit <i>amet</i>, consectetur adipisicing elit.  
A, architecto <strong>asperiores</strong>, aut deleniti deserunt ducimus  
fuga ipsa iure nisi, <span>non optio</span> porro quibusdam quidem  
totam voluptas! <em>Eius error nisi velit</em>.
```

`</p>`



Liens

► Liens

Balise inline, la balise délimite les éléments sur lesquels porte le lien (la zone cliquable).

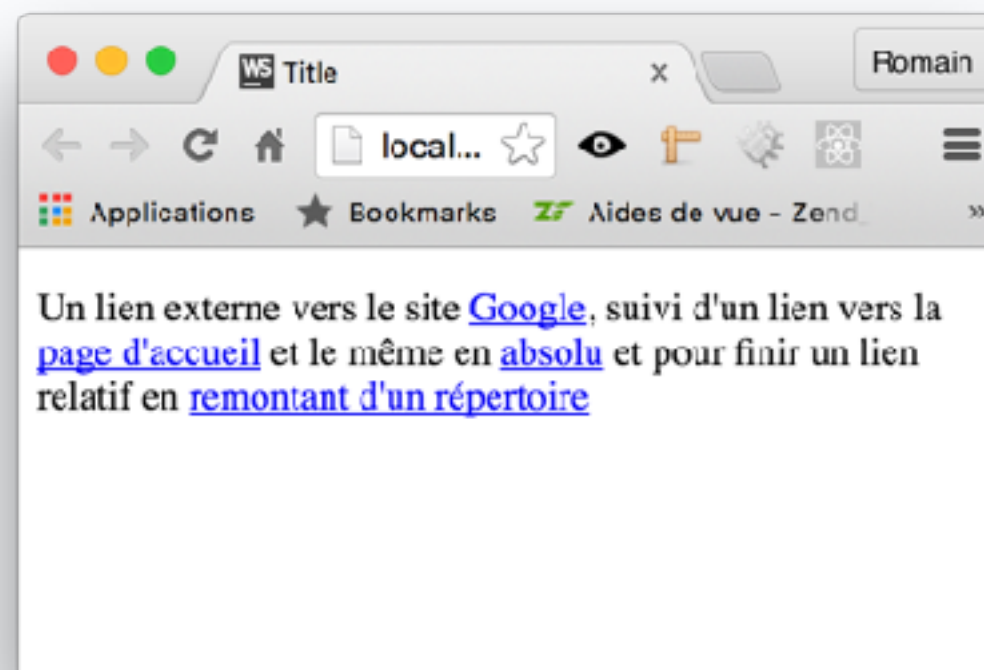
L'attribut href permet de définir la cible du lien. Le lien peut être :

- Externe : commence par http:// ou un autre protocole
- Relatif : dépend d'où se trouve le fichier (../ pour remonter d'un dossier)
- Absolu : depuis la racine du serveur (après http://domaine.ext)

<p>

Un lien externe vers le site Google, suivi d'un lien vers la page d'accueil et le même en absolu et pour finir un lien relatif en remonçant d'un répertoire

</p>



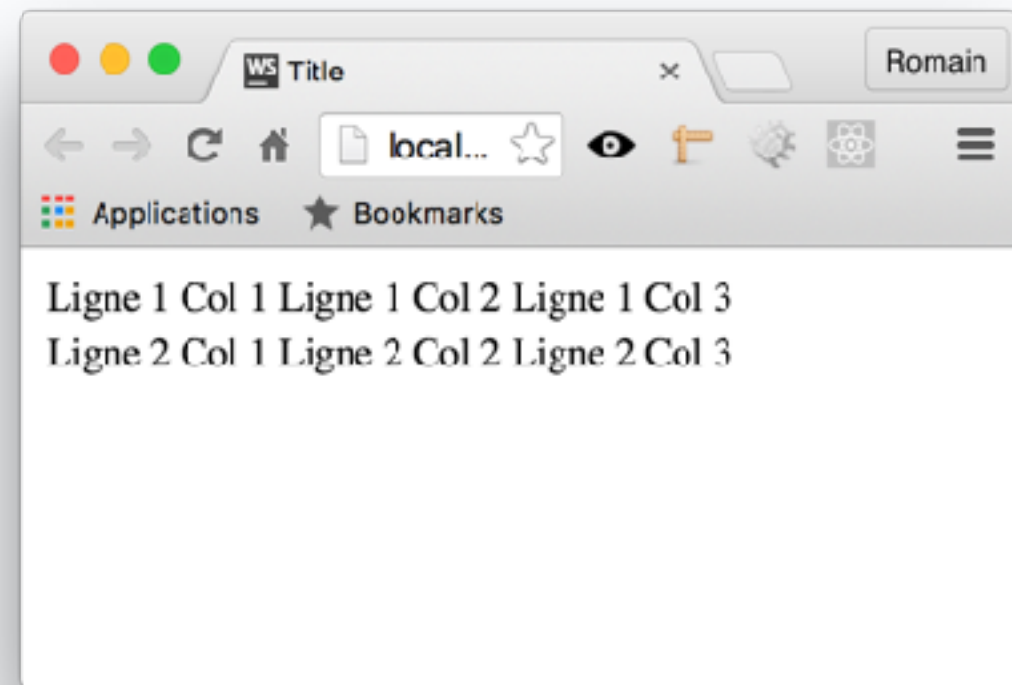
Exercices

- Exercices
 - Créer une page cv.html contenant :
 - Le titre « Description » suivi d'un paragraphe vous présentant
 - Le titre « Photo » suivi d'une photo
 - Le titre « Loisirs » suivi d'une liste de loisirs
 - Sur la page index.html ajouter un lien vers la page cv.html

Tableaux

- Les tableaux sont utilisés pour présenter des données par lignes ou colonnes (penser tableau Excel)
- 3 balises principales, table pour l'ensemble du tableau, tr pour les lignes et td pour les cellules

```
<table>  
  <tr>  
    <td>Ligne 1 Col 1</td>  
    <td>Ligne 1 Col 2</td>  
    <td>Ligne 1 Col 3</td>  
  </tr>  
  <tr>  
    <td>Ligne 2 Col 1</td>  
    <td>Ligne 2 Col 2</td>  
    <td>Ligne 2 Col 3</td>  
  </tr>  
</table>
```



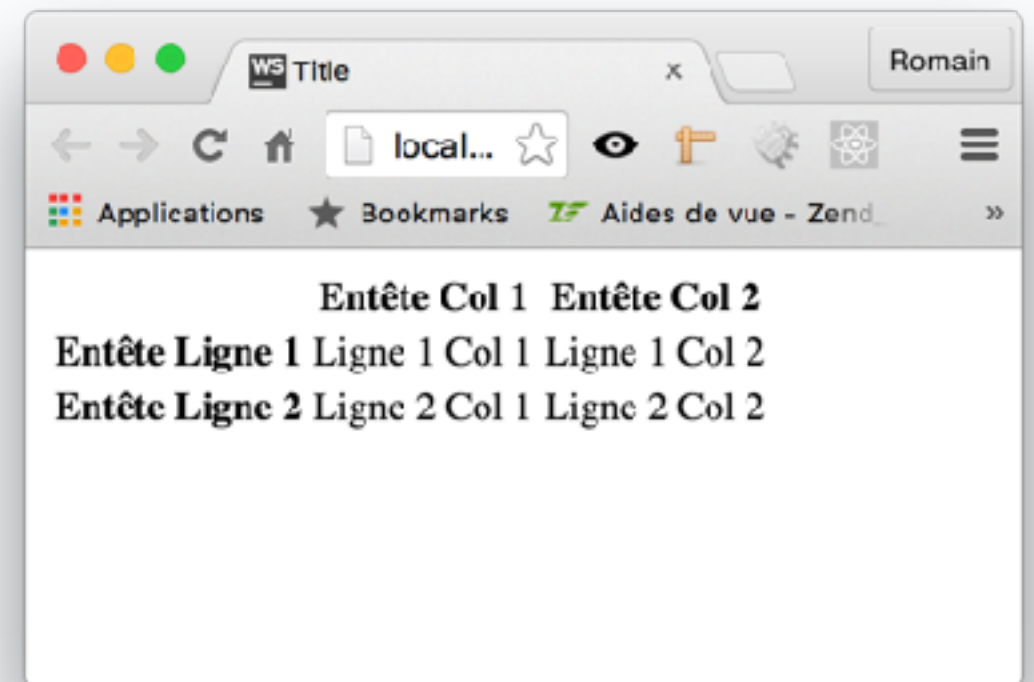
Tableaux

► Les entêtes

Il est possible de définir des cellules d'entêtes avec la balise `th`

L'attribut `scope` permet via les valeurs « `col` » ou « `row` » de spécifier s'il s'agit respectivement d'un entête de ligne ou de colonne.

```
<table>
  <tr>
    <th></th>
    <th scope="col">Entête Col 1</th>
    <th scope="col">Entête Col 2</th>
  </tr>
  <tr>
    <th scope="row">Entête Ligne 1</th>
    <td>Ligne 1 Col 1</td>
    <td>Ligne 1 Col 2</td>
  </tr>
  <tr>
    <th scope="row">Entête Ligne 2</th>
    <td>Ligne 2 Col 1</td>
    <td>Ligne 2 Col 2</td>
  </tr>
</table>
```

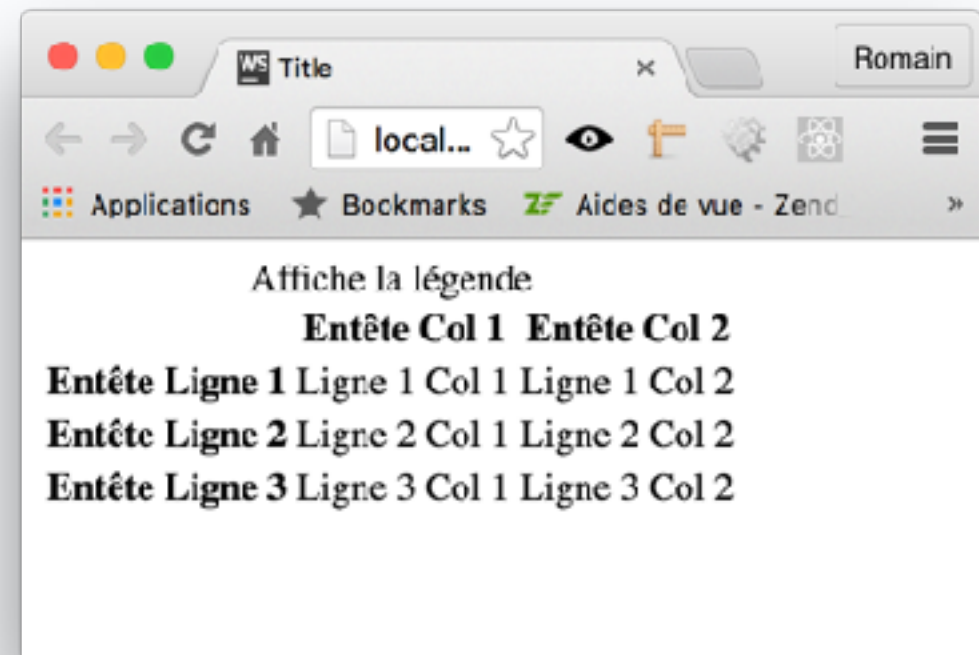


Tableaux

▸ Détailler un tableau

- thead, tbody, tfoot : pour regrouper les lignes d'entêtes, de corps ou de pied.
- summary et caption : légende (seule caption s'affiche)

```
<table summary="Un tableau avec des sections">
  <caption>Affiche la légende</caption>
  <thead>
    <tr>
      <th></th>
      <th scope="col">Entête Col 1</th>
      <th scope="col">Entête Col 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="row">Entête Ligne 1</th>
      <td>Ligne 1 Col 1</td>
      <td>Ligne 1 Col 2</td>
    </tr>
    <tr>
      <th scope="row">Entête Ligne 2</th>
      <td>Ligne 2 Col 1</td>
      <td>Ligne 2 Col 2</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <th scope="row">Entête Ligne 3</th>
      <td>Ligne 3 Col 1</td>
      <td>Ligne 3 Col 2</td>
    </tr>
  </tfoot>
</table>
```

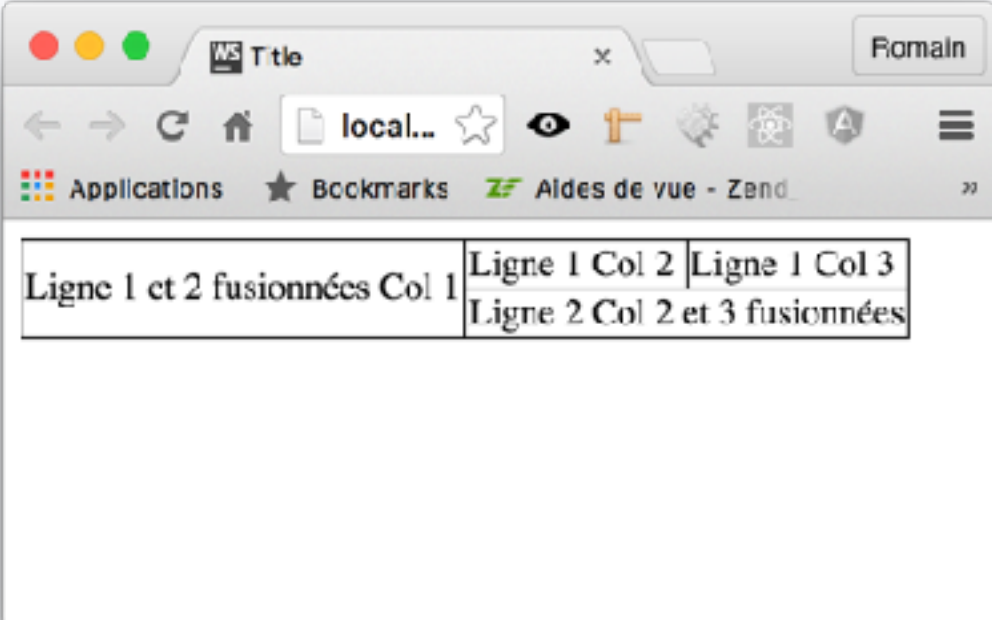


Tableaux

► Fusionner des cellules

- l'attribut `rowspan` permet de fusionner avec les cellules du dessous (le nombre indique le nombre total de cellules à fusionner)
- l'attribut `colspan` permet la fusion de colonnes

```
<table>
  <tr>
    <td rowspan="2">Ligne 1 et 2
fusionnées Col 1</td>
    <td>Ligne 1 Col 2</td>
    <td>Ligne 1 Col 3</td>
  </tr>
  <tr>
    <td colspan="2">Ligne 2 Col 2
et 3 fusionnées</td>
  </tr>
</table>
```



The screenshot shows a web browser window with the title "Title" and a user profile "Romain". The address bar shows "local...". The browser's toolbar includes navigation buttons, a search bar, and icons for applications, bookmarks, and view aids. The rendered table is displayed below the toolbar.

Ligne 1 et 2 fusionnées Col 1	Ligne 1 Col 2	Ligne 1 Col 3
Ligne 2 Col 2 et 3 fusionnées		

Exercices

- ▶ Exercices
 - Créer une page timetable contenant le tableau suivant :

Time Table					
Hours	Mon	Tue	Wed	Thu	Fri
	Science	Maths	Science	Maths	Arts
	Social	History	English	Social	Sports
	Lunch				
	Science	Maths	Science	Maths	Project
	Social	History	English	Social	

Multimédia

▸ Images

```

```

▸ Audio

```
<audio controls>  
  <source src="fichier.ogg" type="audio/ogg">  
  <source src="fichier.mp3" type="audio/mpeg">  
  Balise audio non-soutenue  
</audio>
```

▸ Vidéos

```
<video controls>  
  <source src="fichier.ogg" type="video/ogg">  
  <source src="fichier.mp4" type="video/mpeg">  
  Balise video non-soutenue  
</video>
```

https://developer.mozilla.org/fr/docs/Web/HTML/formats_media_support

Formulaires

- ▶ Formulaires

```
<form action="page.php" method="post">  
  <div>  
    Prénom : <input type="text" name="prenom">  
  </div>  
  <div>  
    <button type="submit">Valider</button>  
  </div>  
</form>
```

- ▶ Action : URL de la page à qui seront envoyé les données

- ▶ Method : Type de requête

GET : lorsqu'il y a un intérêt à ce que la page appelée soit mis en favoris (ex : formulaires de recherche)

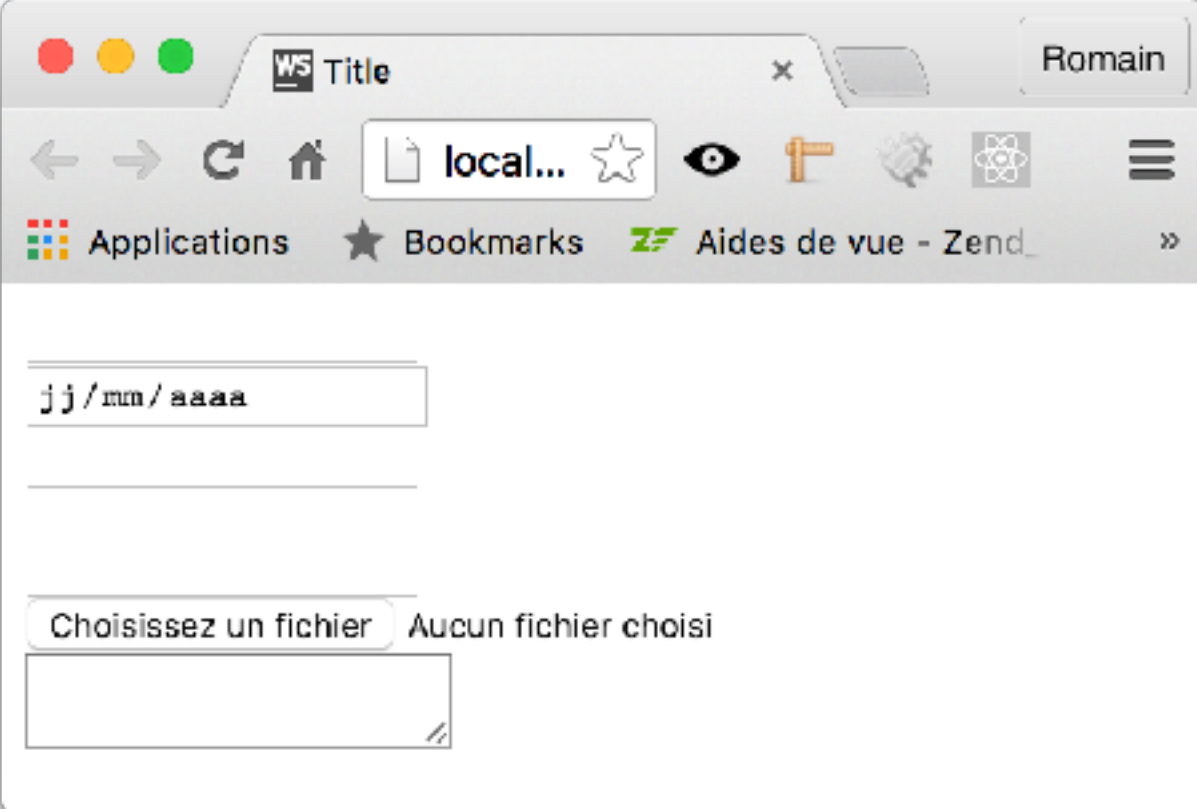
POST : les données ne sont pas dans l'URL donc pas en favoris (inscription, envoi de messages, login/password...)

- ▶ Name : sur chaque champ devant être envoyé au serveur

Formulaires

▸ Type de champs

```
<input type="text">  
<input type="date">  
<input type="email">  
<input type="tel">  
<input type="password">  
  
<input type="hidden">  
  
<input type="file">  
  
<textarea></textarea>
```



A screenshot of a web browser window titled "WS Title" with a user profile "Romain". The browser's address bar shows "local...". The page content includes a form with the following elements:

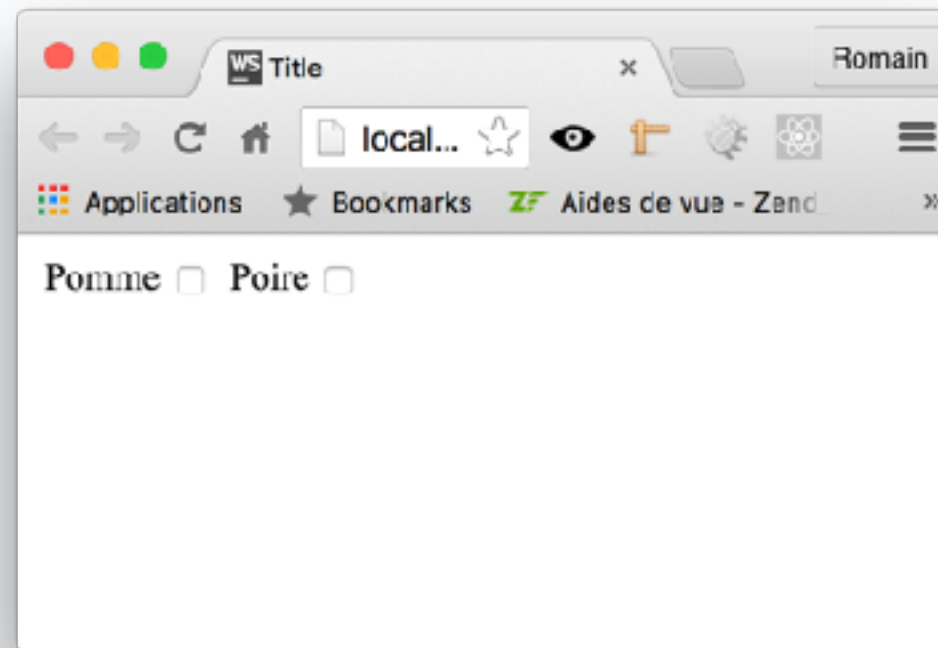
- A date input field containing "jj/mm/aaaa".
- A file input field with the text "Choisissez un fichier" and "Aucun fichier choisi".
- A text area below the file input field.

Formulaires

▸ Cases à cocher

La balise label permet de rendre toute la zone cliquable

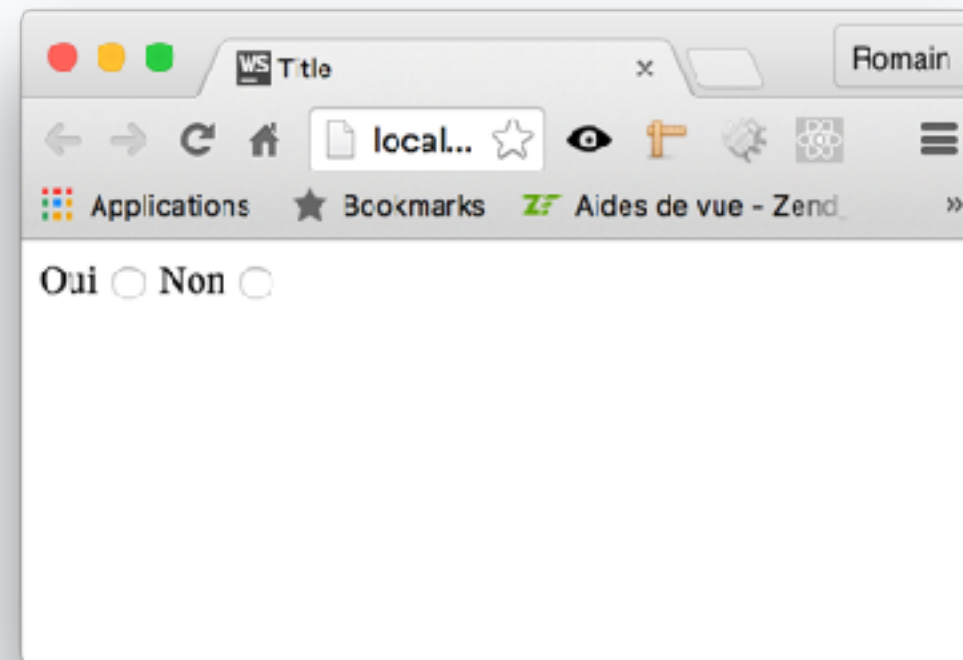
```
<label>
    Pomme
    <input type="checkbox" value="pomme" name="fruit">
</label>
<label>
    Poire
    <input type="checkbox" value="poire" name="fruit">
</label>
```



Formulaires

► Boutons Radio

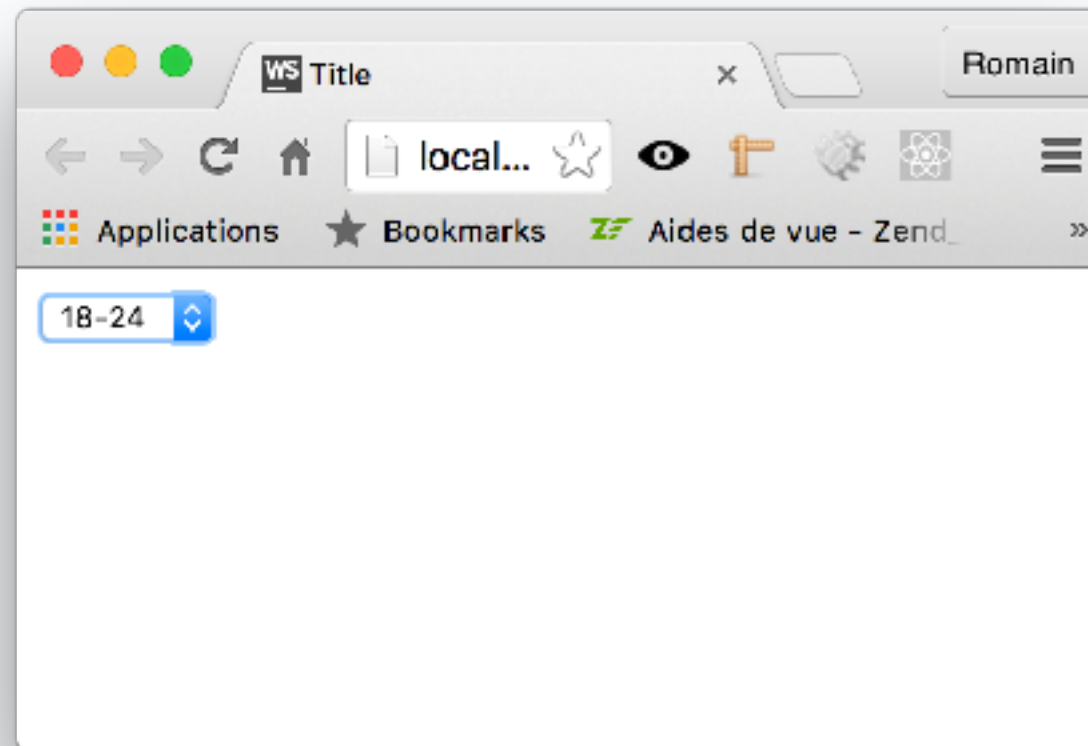
```
<label>
    Oui
    <input type="radio" value="Oui" name="OuiNon">
</label>
<label>
    Non
    <input type="radio" value="Non" name="OuiNon">
</label>
```



Formulaires

- Liste déroulante

```
<select>  
  <option>18-24</option>  
  <option>25-29</option>  
  <option>30-35</option>  
</select>
```



Attributs courants

▸ Identifiants

- id

L'attribut id permet d'identifier de manière unique un élément sur la page (utile pour le CSS et JS)

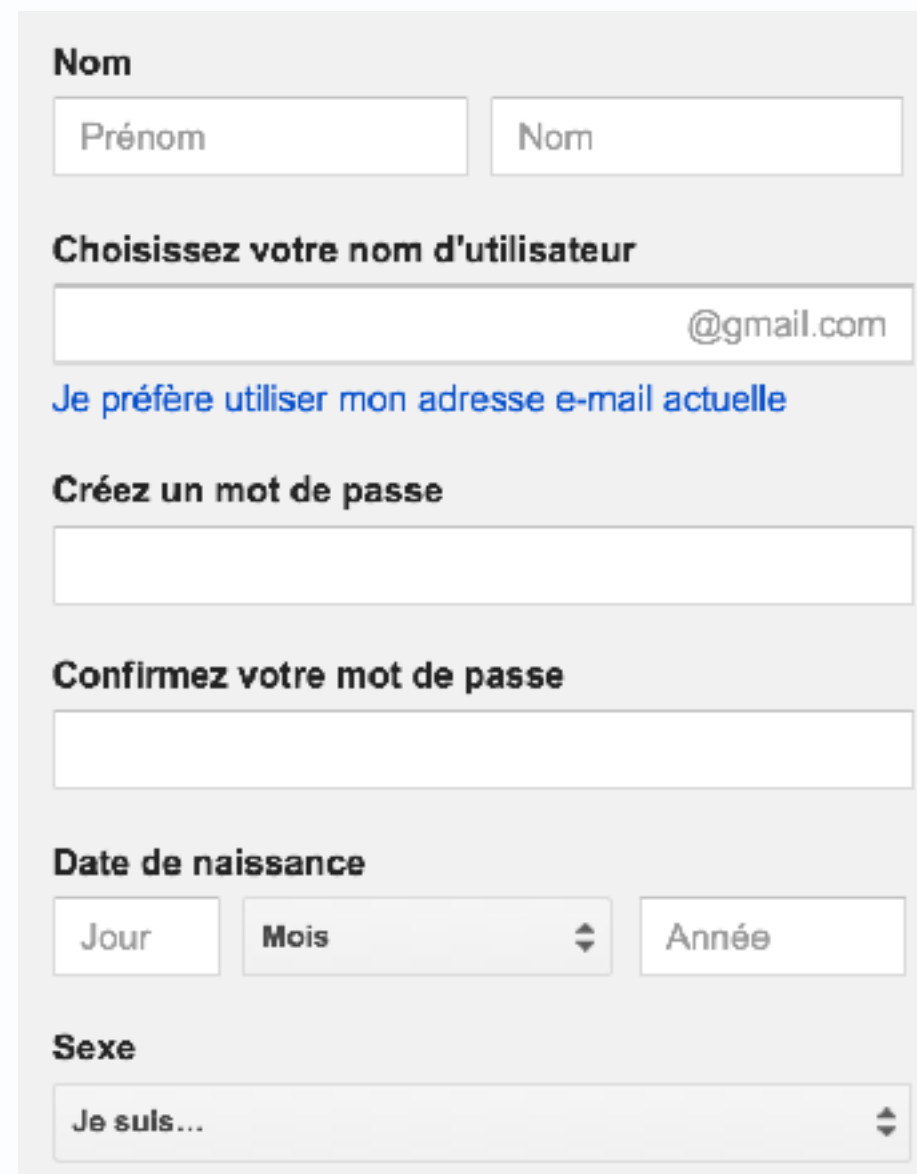
- class

L'attribut class permet d'identifier un groupe d'élément sur la page

```
<ul id="liste-prenoms">  
  <li id="moi" class="prenom francais">Romain</li>  
  <li class="prenom anglais">William</li>  
  <li class="prenom espagnol">Roberto</li>  
</ul>
```

Exercices

- ▶ Exercices
 - Créer une page inscription.html contenant le formulaire suivant



The image shows a registration form with the following fields and labels:

- Nom**: Two input fields, the first labeled "Prénom" and the second labeled "Nom".
- Choisissez votre nom d'utilisateur**: A single input field with a placeholder "@gmail.com".
- Je préfère utiliser mon adresse e-mail actuelle**: A blue link text.
- Créez un mot de passe**: A single input field.
- Confirmez votre mot de passe**: A single input field.
- Date de naissance**: Three input fields labeled "Jour", "Mois", and "Année". The "Mois" field has a dropdown arrow.
- Sexe**: A single input field with a dropdown arrow and the placeholder "Je suis...".

CSS - Sélecteurs

Sélecteurs CSS

- Les sélecteurs CSS est un langage de recherche de noeud dans un arbre
- Ils permettent de sélectionner les éléments de la page (balises) auxquels s'appliqueront des propriétés CSS
- Norme actuelle : Selectors Level 3
<http://www.w3.org/TR/selectors/>
<http://caniuse.com/#feat=css-sel3>
- Norme en cours de redaction : Selectors Level 4
<http://www.w3.org/TR/selectors4/>
<http://css4-selectors.com/browser-selector-test/>

Sélecteurs CSS

▸ Exemple d'utilisation des Sélecteurs CSS

- En CSS

```
#compteur {  
  width: 100px;  
  height: 100px;  
  background: red;  
}
```

- En JavaScript

```
var compteur = document.querySelector('#compteur');  
compteur.addEventListener('click', function() {  
  compteur.innerHTML++;  
});
```

- Avec jQuery

```
var $compteur = $('#compteur');  
$compteur.click(function() {  
  $compteur.toggleClass('inverse');  
});
```

Sélecteurs CSS

► Sélecteurs Basiques

*	tous les éléments	*
E	un élément de type E	div p h1
E.warning	un élément E dont l'une des classes est warning	div.important *.center .center button.btn.btn-primary
E#myid	un élément E dont l'id est myId	div#boite *#boite #boite
E, F	les éléments E et les éléments F	div, p h1, h2, h3, h4, h5, h6 button.btn, a.btn div.center, #box, p

Sélecteurs CSS

► Sélecteurs Hiérarchiques

$E\ F$	un élément F descendant de E	#boite a a span.icon
$E > F$	un élément F fils de E	body > h1 ul > li
$E + F$	un élément F immédiatement précédé par un élément E	h2 + p li + li
$E \sim F$	un élément F précédé par un élément E	h2 ~ form

Sélecteurs CSS

► Sélecteurs par pseudo-classes hiérarchiques

E:root	un élément E, racine du document	html:root :root
E:nth-child(n)	un élément E, nième enfant de son parent	li:nth-child(3) li:nth-child(odd) li:nth-child(2n+1)
E:nth-last-child(n)	un élément E, nième enfant en partant de la fin	li:nth-last-child(3)
E:nth-of-type(n)	un élément E, nième enfant du même type	h2:nth-of-type(3) h2:nth-of-type(odd) h2:nth-of-type(3n)
E:nth-last-of-type(n)	un élément E, nième enfant du même type en partant de la fin	h2:nth-last-of-type(3)
E:first-child	un élément E, premier enfant	li:first-child

Sélecteurs CSS

▸ Sélecteurs par pseudo-classes hiérarchiques (suite)

E:last-child	un élément E, dernier enfant	li:last-child
E:first-of-type	un élément E, premier enfant du même type	h2:first-of-type
E:last-of-type	un élément E, dernier enfant du même type	h2:last-of-type
E:only-child	un élément E, fils unique	div:only-child
E:only-of-type	un élément E, fils unique du même type	form:only-of-type
E:empty	un élément E, qui n'a pas d'enfant, excepté des noeuds de texte	div:empty

Sélecteurs CSS

► Sélecteurs par attributs

E[foo]	un élément E avec un attribut foo	*[id] [id]
E[foo="bar"]	un élément E dont l'attribut foo vaut bar	img[src="monimage.png"]
E[foo~="bar"]	un élément E dont l'attribut foo contient le mot bar	a[rel~="external"] button[class~="btn"]
E[foo^="bar"]	un élément E dont l'attribut foo commence par bar	a[href^="https://"]
E[foo\$="bar"]	un élément E dont l'attribut foo finit par bar	img[src\$=".png"]
E[foo*="bar"]	un élément E dont l'attribut foo contient bar	div[id*="box-"]
E[foo ="en"]	un élément E dont l'attribut foo contient en ou en- suivi de quelque chose	a[hreflang "en"]

Sélecteurs CSS

▸ Sélecteurs par pseudo-classes de contexte

E:link	un lien non visité	a:link
E:visited	un lien visité	a:visited
E:active	un élément actif (clic en cours)	a:active div:active
E:hover	un élément survolé	a:hover div:hover
E:focus	un élément qui reçoit le focus	input:focus

Sélecteurs CSS

► Sélecteurs par pseudo-classes

E:target	un élément cible d'un lien	div:target
E:lang(fr)	un élément dont la langue est fr	div:lang(fr)
E:enabled	un élément activé	input:enabled
E:disabled	un élément désactivé	input:disabled
E:checked	un élément coché	input[type="checkbox"]:checked
E:not(s)	un élément E qui ne correspond pas au sélecteur simple s	div:not(.important)

Sélecteurs CSS












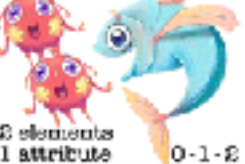








▸ Sélecteurs par pseudo-élément

E::first-line	la premier ligne de l'élément E	p::first-line
E::first-letter	la premier lettre de l'élément E	p::first-letter
E::before	contenu généré avant l'élément E	div::before
E::after	contenu généré après l'élément E	div::after

Sélecteurs CSS

- Priorités

<http://www.standardista.com/css3/css-specificity/>

CSS SPECIFICITY			
WITH PLANKTON, FISH AND SHARKS			
*  universal selector 0-0-0	div  1 element 0-0-1	li > ul  2 elements 0-0-2	body div ... ul li p a  12 elements 0-0-12
.myClass  1 class 0-1-0	*.myClass  1 universal selector 1 class 0-1-0	[type=checkbox]  1 attribute selector 0-1-0	:only-of-type  1 pseudo-class 0-1-0
li.myClass  1 element 1 class 0-1-1	li[attr]  1 element 1 attribute 0-1-1	li:nth-of-type(3n)~li  3 elements 1 pseudo-class 0-1-2	form input[type=email]  2 elements 1 attribute 0-1-2
li.class:nth-of-type(3n)  1 element 1 class 1 pseudo-class 0-2-1	input[type].not(.class)  1 element 1 class 1 attribute 0-2-1	li:nth-child(3n)~li[type]...  10 elements 1 pseudo-class 0-10-0	#myDiv  ID Selector 1-0-0
#myDiv li.class a[href]  2 types 2 class/attribute 1 ID Selector 1-2-2	#divitis #myDiv a  2 ID Selectors 1 type selector 2-0-1	style=""  inline style 1-0-0-0	!important  Important 0-0-0-0

X-0-0: The number of ID selectors

0-Y-0: The number of class selectors, attributes selectors, and pseudo-classes

0-0-Z: The number of type selectors and pseudo-elements

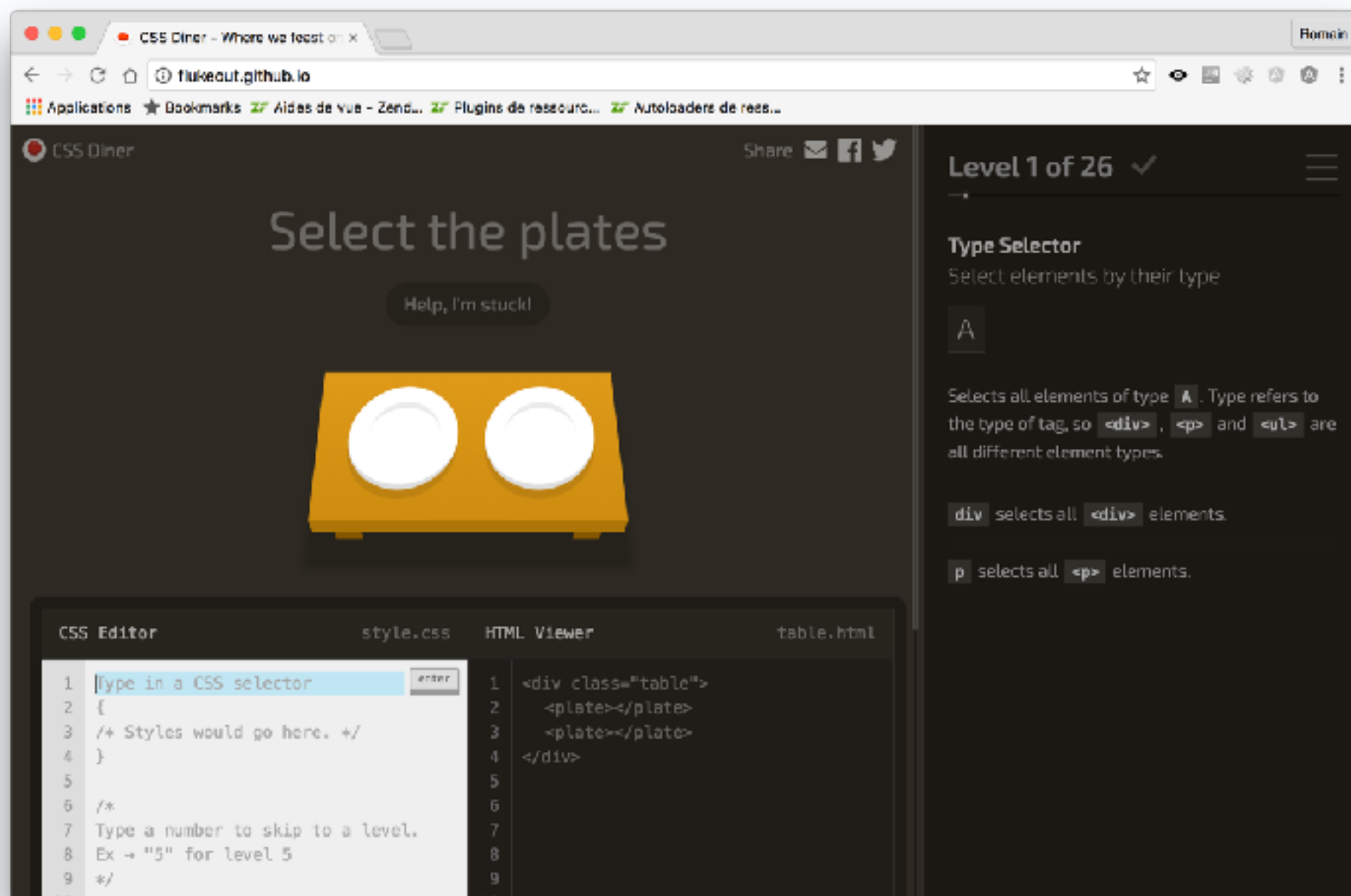
*, +, >, ~: Universal selector and combinators do not increase specificity

:not(x): Negation selector has no value. Argument increases specificity



Sélecteurs CSS

- Apprendre en s'amusant
<http://flukeout.github.io/>



CSS - Modèle de boîte

Modèle de boîte CSS

- ▶ **Modèle de boîte CSS**

Chaque élément d'un document est matérialisé par une boîte qui peut être ajustée grâce à des propriétés CSS spécifiques.

- **Content-box**

Boîte du contenu

- **Padding-box**

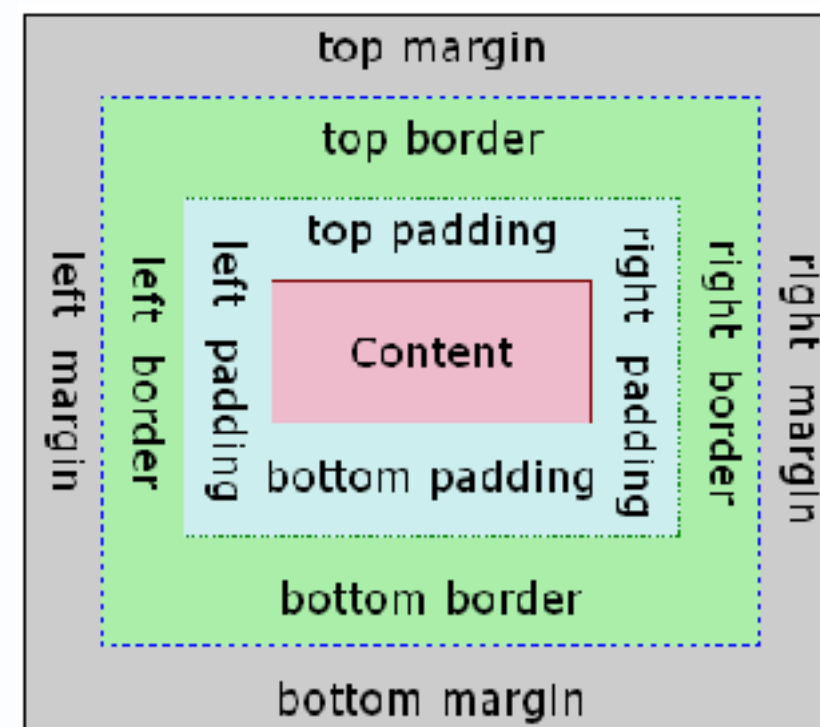
Marge interne (inclus le background)

- **Border-box**

Bordure

- **Margin-box**

Marge externe



Modèle de boîte CSS

- **box-sizing**

- **content-box**

Par défaut, la hauteur totale d'un élément sera de :

margin-top + border-top + padding-top + height + padding-bottom + border-bottom + margin-bottom

- **border-box**

En box-sizing border-box, la hauteur de l'élément inclus le padding et la bordure, ainsi la hauteur total sera de :

margin-top + border-top + padding-top + height + padding-bottom + border-bottom + margin-bottom

- (idem pour le calcul de la largeur dans les 2 cas)

- Les calculs étant simplifiées en box-sizing border-box, certains bibliothèques comme Bootstrap l'activent pour tous les éléments :

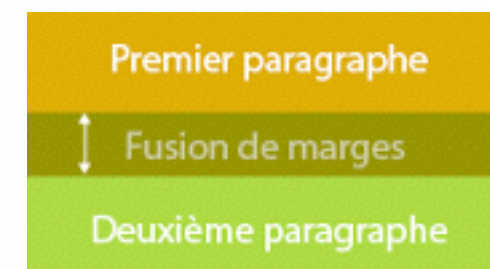
```
* {  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}
```

Modèle de boîte CSS

- ▶ **Fusion des marges**

Les marges haute et basse des blocs sont parfois fusionnées en une seule marge dont la taille est la plus grande des deux marges fusionnées. C'est ce qu'on appelle la fusion des marges.

- ▶ **3 cas possibles**



- **Des éléments voisins adjacents**

La marge inférieure d'un élément fusionne avec la marge supérieure de l'élément suivant (ex : les paragraphes).

- **Un élément parent et son premier/dernier élément fils**

La marge supérieure du parent fusionne avec la marge supérieure de ses descendants (si le parent n'a pas de padding et de border), idem pour les marges inférieures

- **Des blocs vides**

Les marges du haut et du bas fusionnent avec les précédents suivants (si pas de padding et de border)

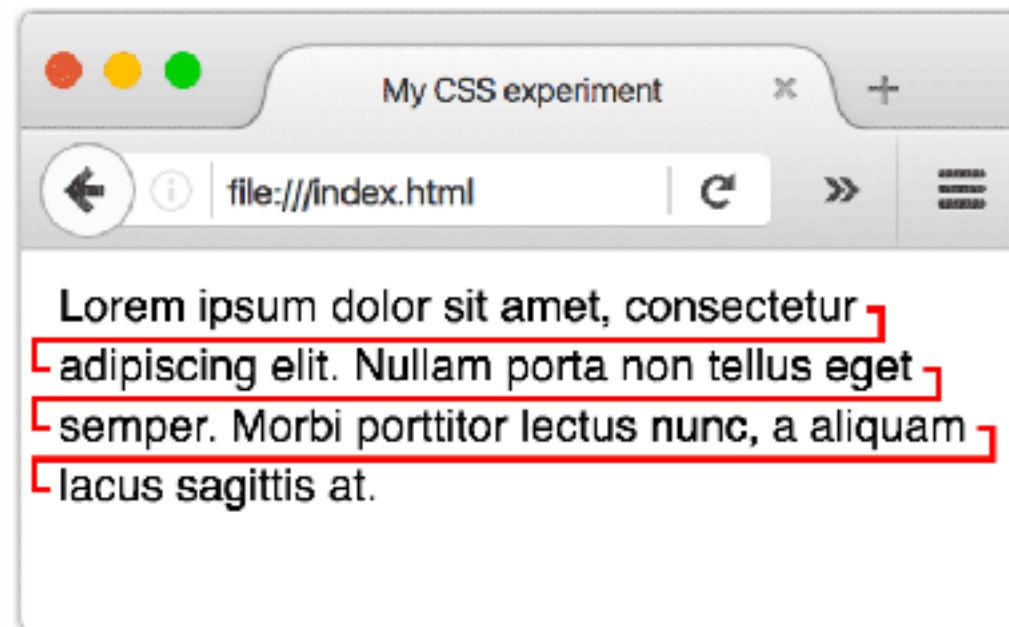
CSS - Positionnement

Positionnement CSS

► Flux

Les balises HTML se positionnent dans leur ordre d'arrivée, de gauche à droite, puis de haut en bas s'il n'y a plus de place sur la ligne (comme des mots au sein d'un texte).

Certaines balises peuvent forcer le passage à la ligne.



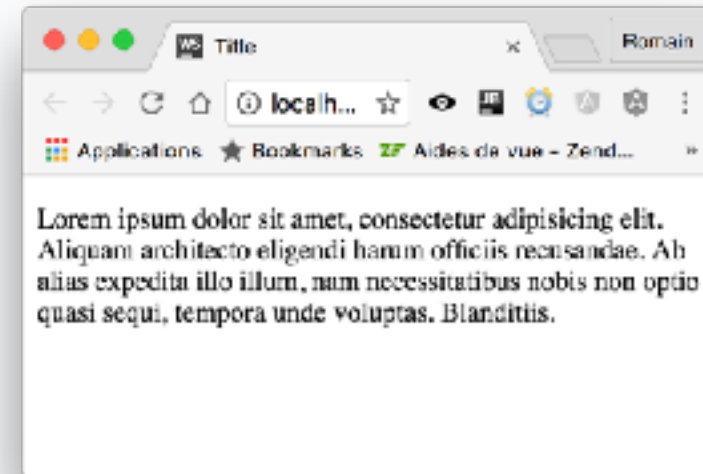
Positionnement CSS

► Gestion des espaces

Le navigateur supprime les espaces et retour à la ligne qui précèdent le premier mot, idem après ceux du dernier mots.

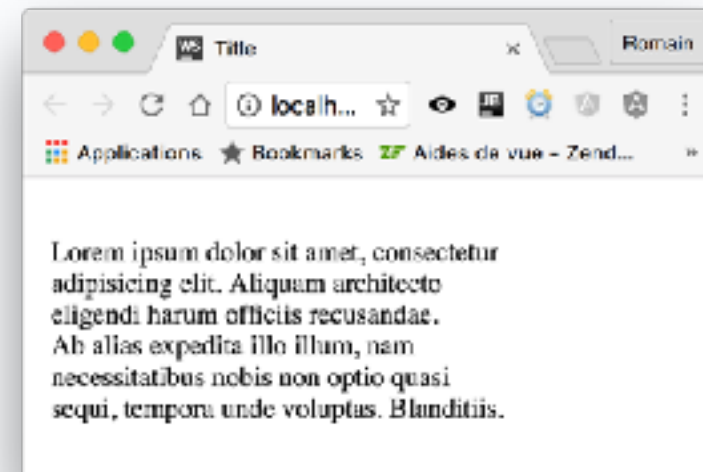
Entre 2 mots, plusieurs espaces ou retour à la ligne sont transformés en un seul espace.

```
<p>
  Lorem ipsum dolor sit amet, consectetur
  adipiscing elit. Aliquam architecto
  eligendi harum officiis recusandae.
  Ab alias expedita illo illum, nam
  necessitatibus nobis non optio quasi
  sequi, tempora unde voluptas. Blanditiis.
</p>
```



► Il est possible de préserver les espaces avec la propriété white-space (comportement par défaut des balises pre et textarea) :

```
p {
  white-space: pre;
}
```



Positionnement CSS

▸ Display

Il y a 4 valeurs courantes pour la propriété display.

- **none**

Cette valeur retire l'élément du flux, comme si l'élément et son contenu n'existaient pas.

- **inline**

Cette valeur rend l'élément transparent au sens où il s'inscrit dans le flux de texte global, il est donc associé au texte l'environnant. Il n'est pas possible de modifier sa largeur et sa hauteur.

Comportement par défaut des balises : a, span, b, i, strong, em...

- **block**

Cette valeur cassera le flux de texte pour insérer l'élément. Cela provoquera donc un saut de ligne avant et après. Le contenu de cet élément ne fait donc pas partie du flux global et suit donc les contraintes de l'élément définies par le modèle de boîte. Largeur par défaut 100%, hauteur auto.

Comportement par défaut des balises : div, p, h1, h2, h3, form, header, footer...

- **inline-block**

Cette valeur est en quelque sorte un intermédiaire entre inline et block. Comme avec inline, les boîtes seront placées dans le flux global mais, comme avec block, le contenu ne fera pas partie du texte environnant. Il est possible de modifier sa largeur et sa hauteur.

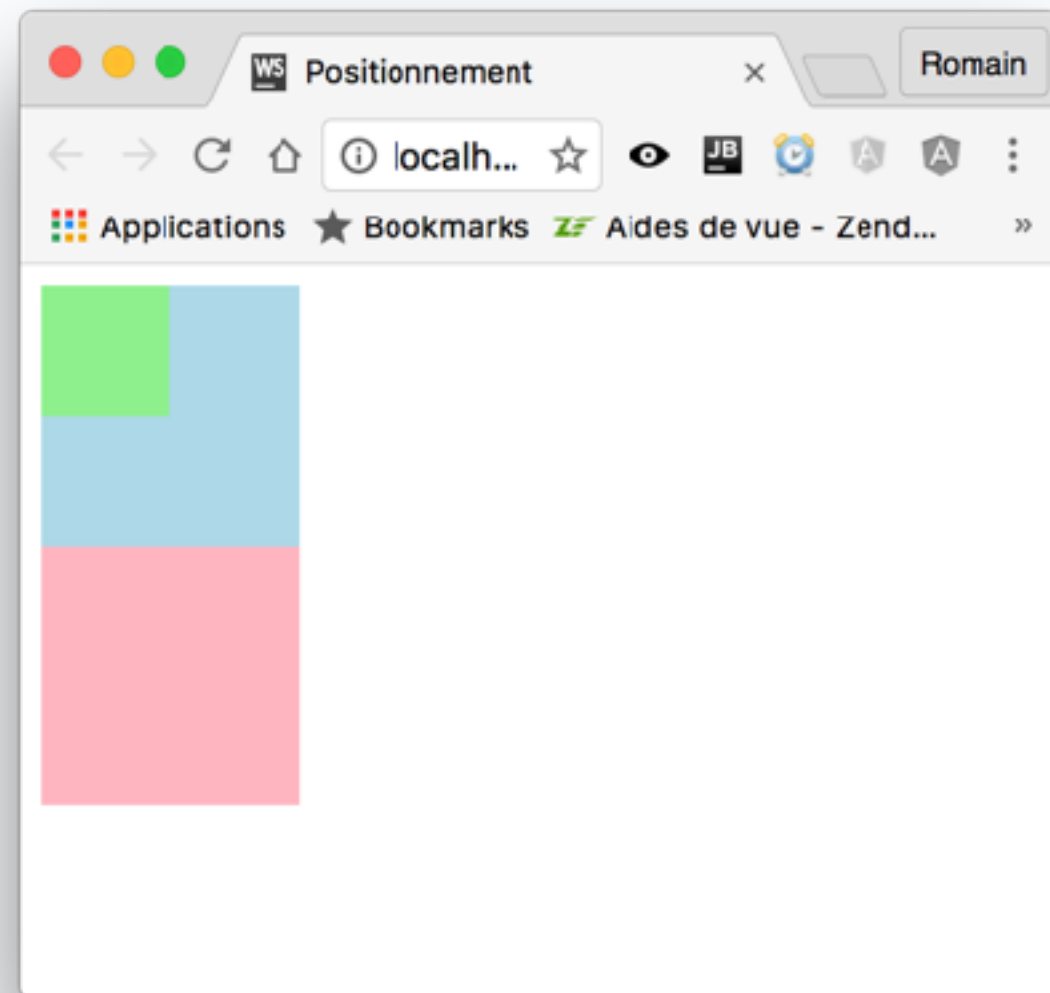
Positionnement CSS

- Soit le HTML + CSS suivant

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Position Static</title>
  <style>
    .externe {
      width: 100px;
      height: 100px;
      background-color: lightblue;
    }

    .interne {
      width: 50px;
      height: 50px;
      background-color: lightgreen;
    }

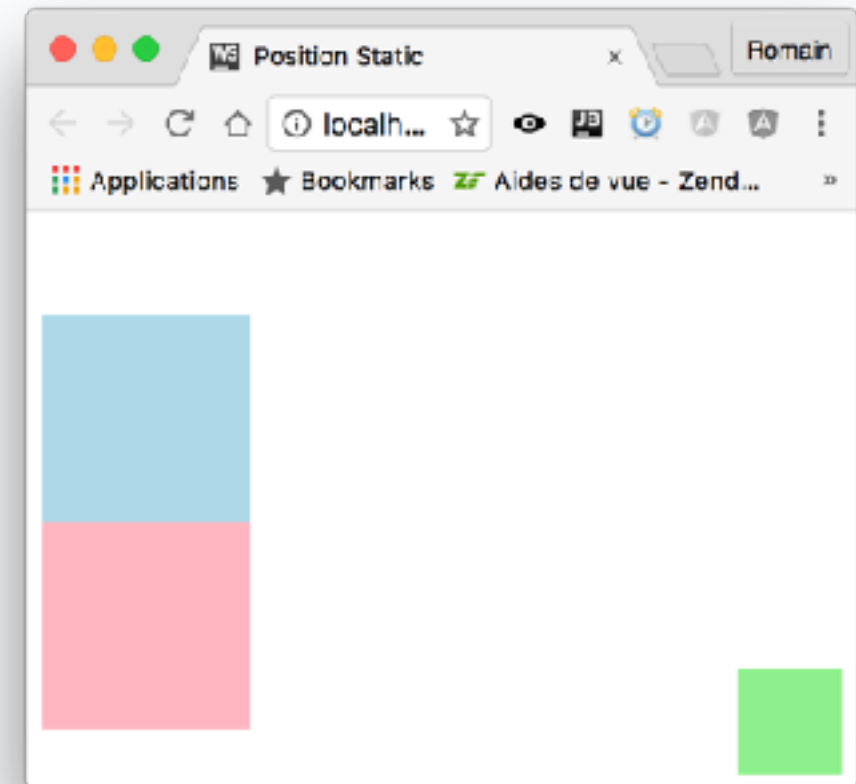
    .suivant {
      width: 100px;
      height: 100px;
      background-color: lightpink;
    }
  </style>
</head>
<body>
  <div class="externe">
    <div class="interne"></div>
  </div>
  <div class="suivant"></div>
</body>
</html>
```



Positionnement CSS

► Positionnement static (valeur par défaut)

```
.externe {  
  position: static; /* valeur par défaut */  
  margin-top: 50px;  
  top: 20px; /* inactif */  
  left: 20px; /* inactif */  
}  
  
.interne {  
  position: absolute;  
  bottom: 8px;  
  right: 8px;  
}
```



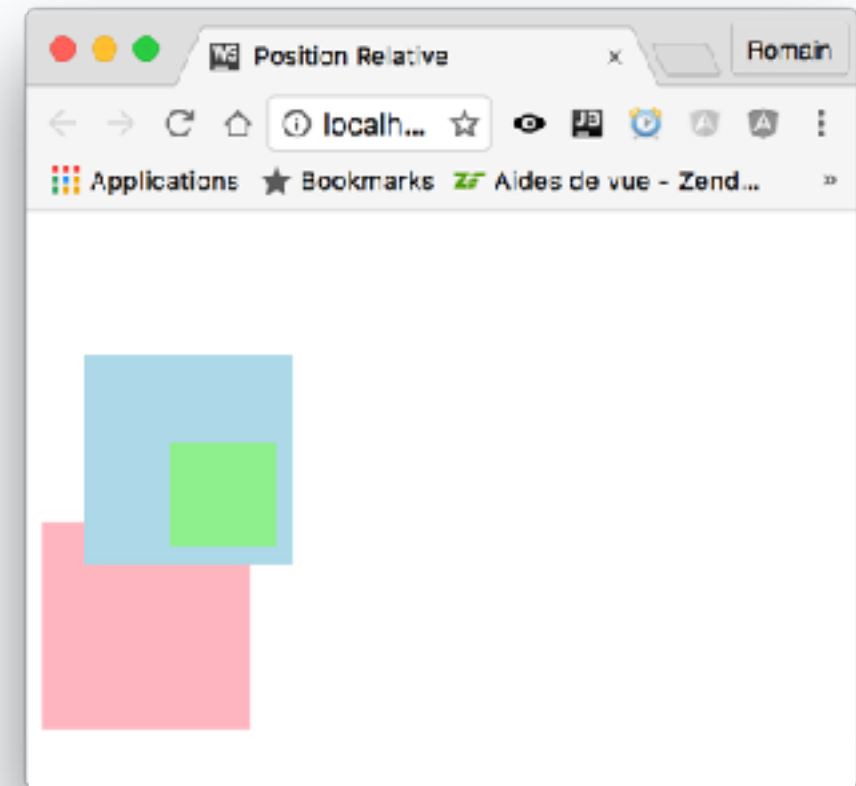
► Remarques :

- positionnement dans le flux, sa taille et ses marges impactent la suite du flux
- top, left, bottom, right inactifs
- l'élément interne (position absolu) s'est positionné par rapport à la page

Positionnement CSS

► Positionnement relative

```
.externe {  
  position: relative;  
  margin-top: 50px;  
  top: 20px;  
  left: 20px;  
}  
  
.interne {  
  position: absolute;  
  bottom: 8px;  
  right: 8px;  
}
```



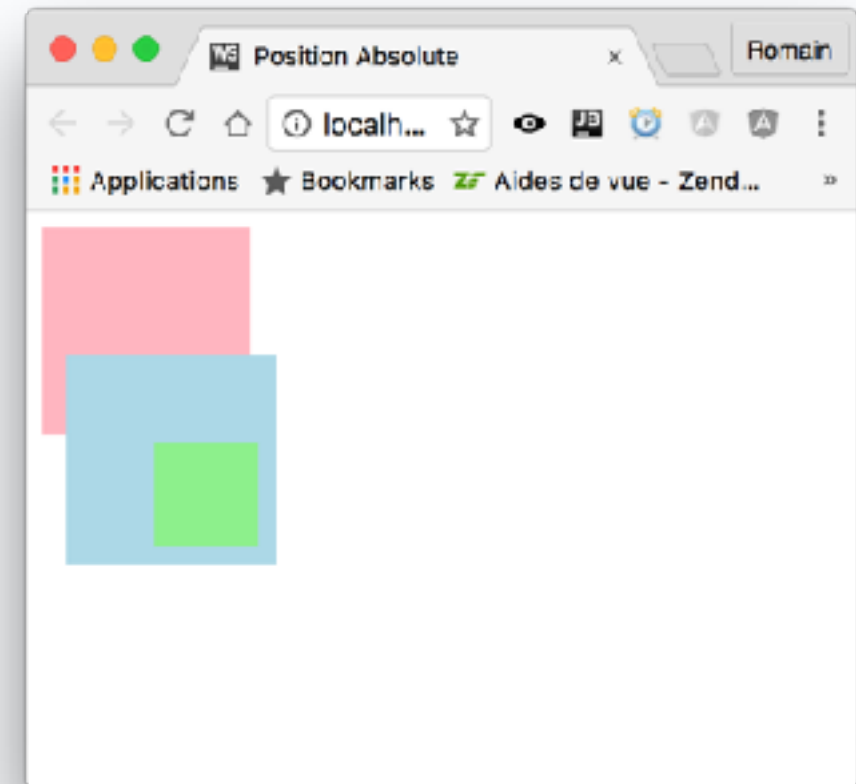
► Remarques :

- positionnement dans le flux, sa taille et ses marges impactent la suite du flux
- top, left, bottom, right déplacent l'élément par rapport à sa position d'origine, sans impacter la suite du flux
- l'élément interne (position absolu) s'est positionné par rapport à son ancêtre non static le plus proche

Positionnement CSS

► Positionnement absolute

```
.externe {  
  position: absolute;  
  margin-top: 50px;  
  top: 20px;  
  left: 20px;  
}  
  
.interne {  
  position: absolute;  
  bottom: 8px;  
  right: 8px;  
}
```



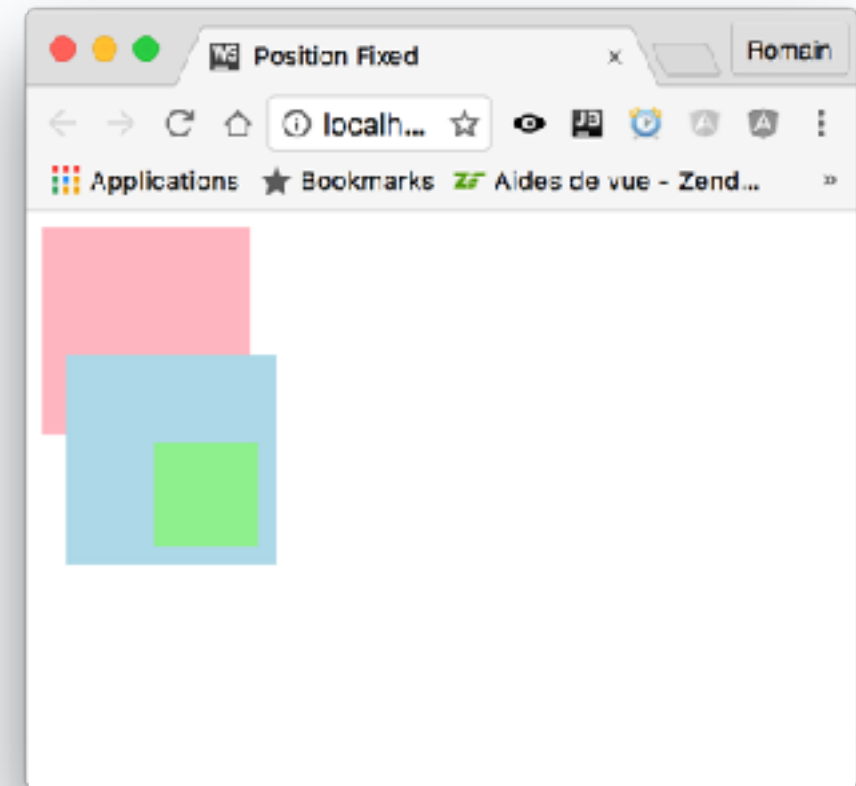
► Remarques :

- positionnement hors flux, les éléments suivants se positionnent comme cet élément n'existait pas
- top, left, bottom, right place l'élément par rapport à la page (si que des ancêtres static)
- l'élément interne (position absolu) s'est positionné par rapport à son ancêtre non static le plus proche

Positionnement CSS

► Positionnement fixed

```
.externe {  
  position: fixed;  
  margin-top: 50px;  
  top: 20px;  
  left: 20px;  
}  
  
.interne {  
  position: absolute;  
  bottom: 8px;  
  right: 8px;  
}
```



► Remarques :

- positionnement hors flux, les éléments suivants se positionnent comme cet élément n'existait pas
- top, left, bottom, right placent l'élément par rapport à la fenêtre
- l'élément interne (position absolu) s'est positionné par rapport à son ancêtre non static le plus proche

Positionnement CSS

- ▶ Float

Permet d'encadrer du texte autour d'un élément.

- ▶ Clear

On utilise la propriété clear pour qu'un élément arrête d'encadrer le texte. Il est également possible d'utiliser overflow sur un élément conteneur.



Positionnement CSS

- Placer 2 éléments block horizontalement

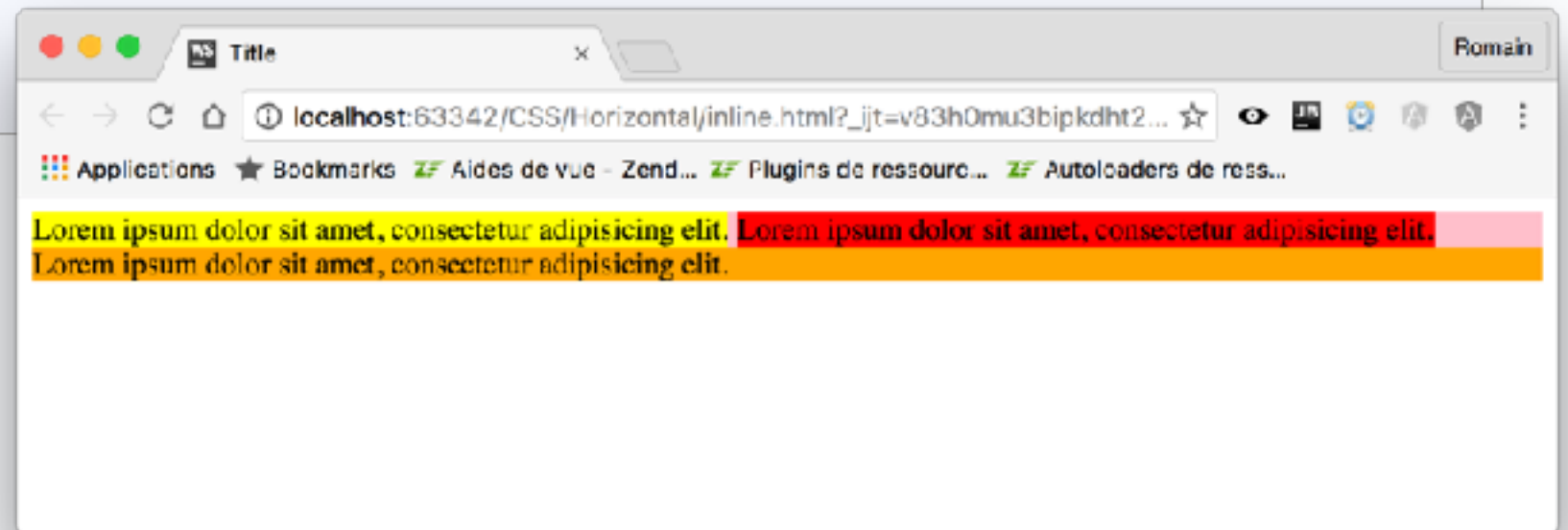
```
<div class="container pink">  
  <div class="col yellow">Lorem ipsum dolor sit amet, consectetur adipisicing elit.</div>  
  <div class="col red">Lorem ipsum dolor sit amet, consectetur adipisicing elit.</div>  
</div>  
<div class="next orange">Lorem ipsum dolor sit amet, consectetur adipisicing elit.</div>
```



Positionnement CSS

- Placer 2 éléments block horizontalement

```
.col {  
  display: inline;  
}
```

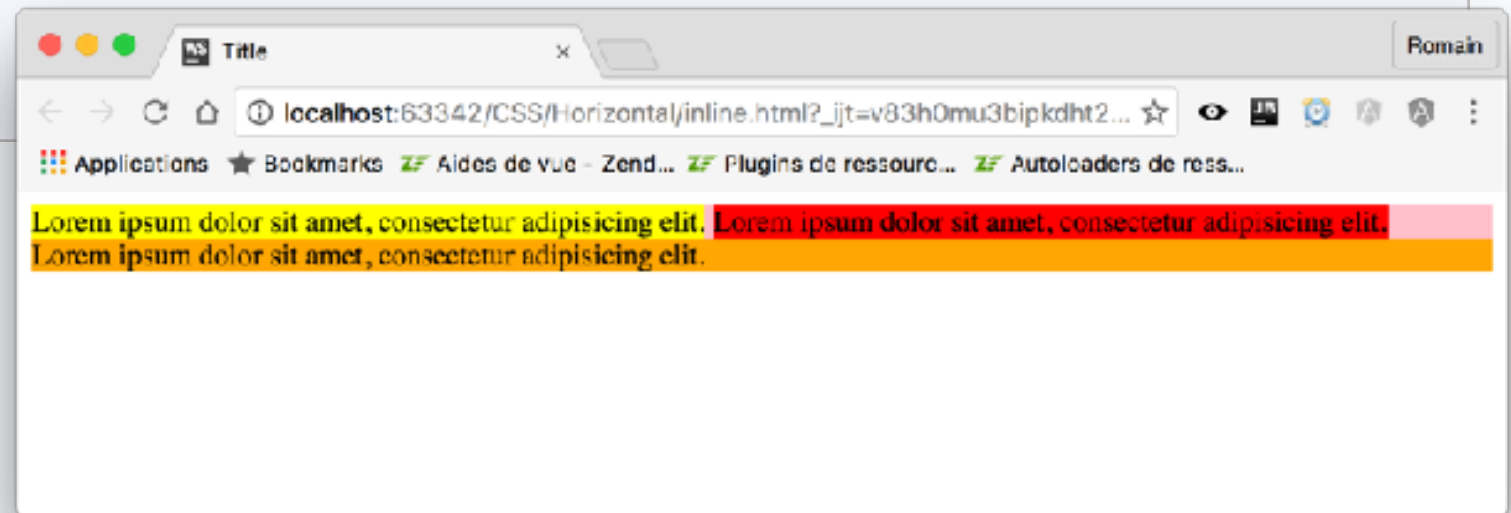


- Pas possible de modifier la largeur et hauteur
- Espace les 2 colonnes

Positionnement CSS

- Placer 2 éléments block horizontalement

```
.col {  
  display: inline-block;  
}
```

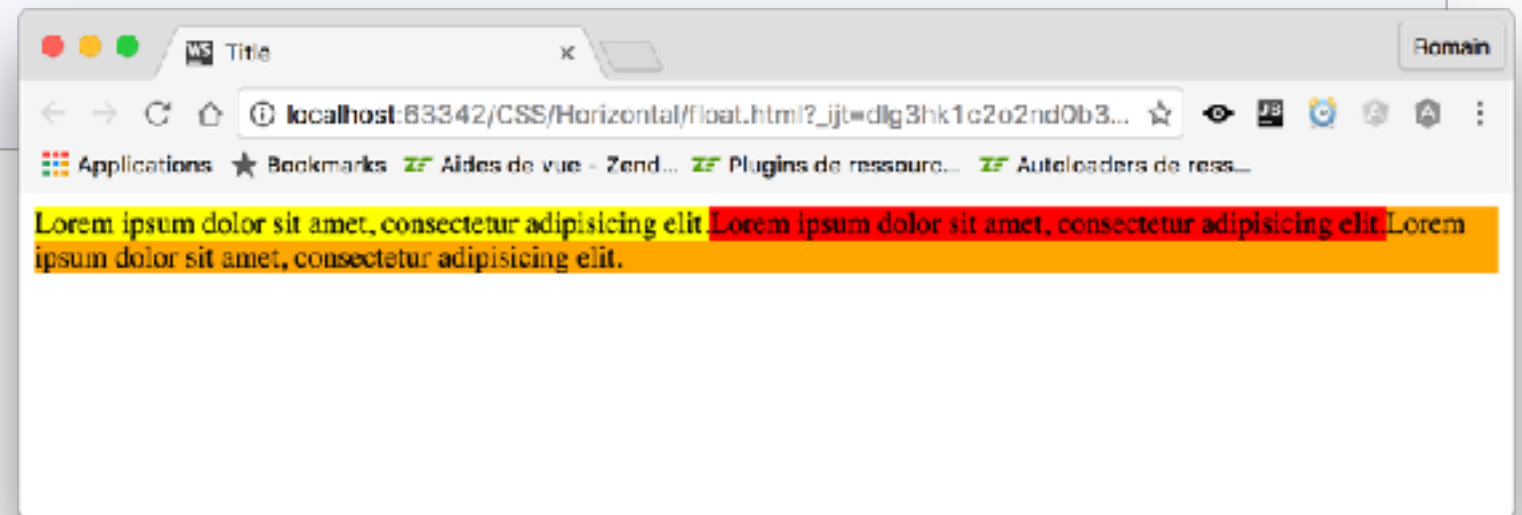


- Espace les 2 colonnes

Positionnement CSS

- Placer 2 éléments block horizontalement

```
.col {  
  float: left;  
}
```

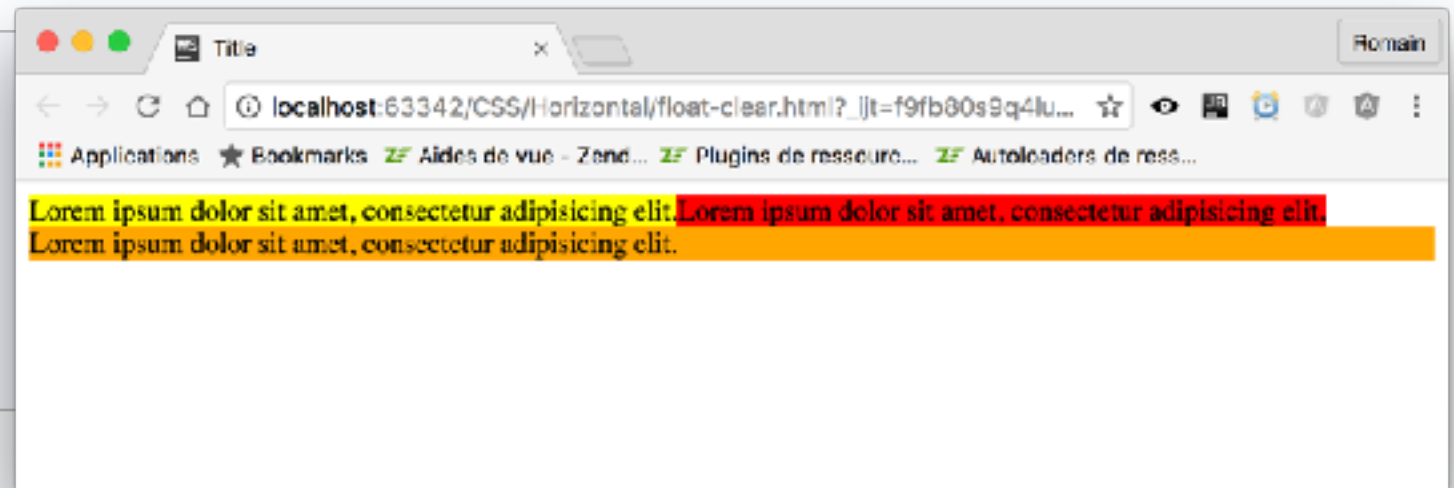


- Le texte suivant encadre les 2 autres

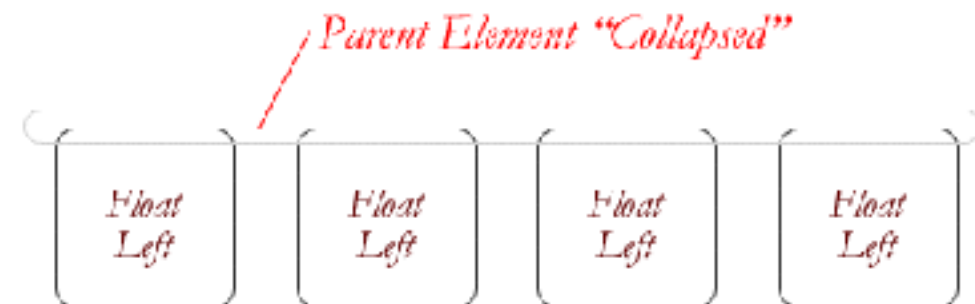
Positionnement CSS

- Placer 2 éléments block horizontalement

```
.col {  
  float: left;  
}  
  
.next {  
  clear: left;  
}
```



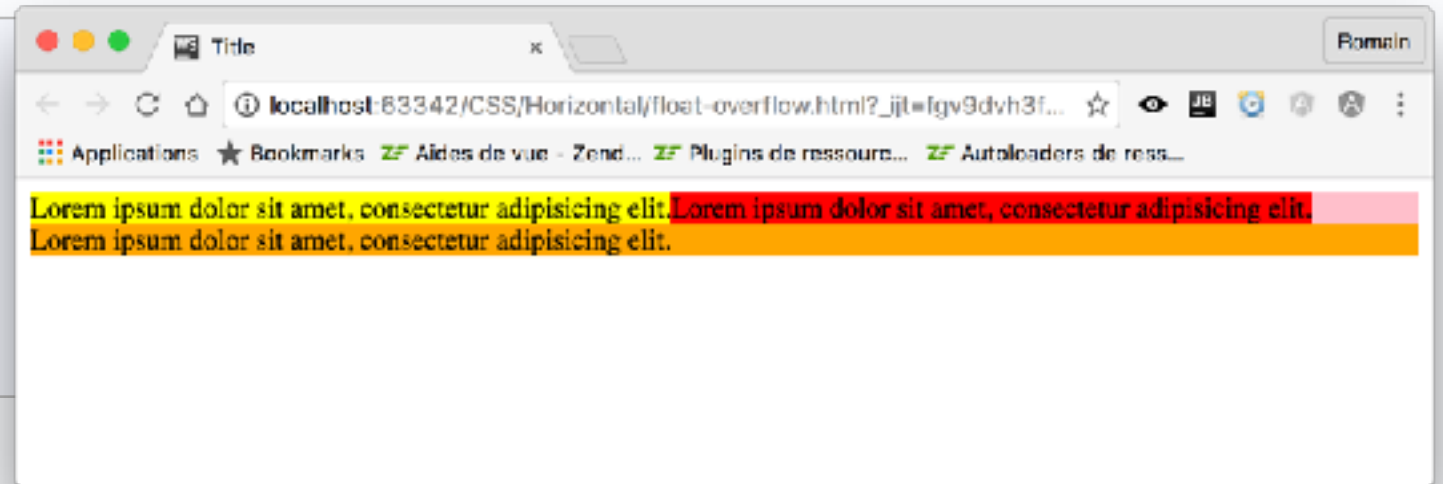
- La hauteur du conteneur est de 0.



Positionnement CSS

- Placer 2 éléments block horizontalement

```
.container {  
  overflow: hidden;  
}  
  
.col {  
  float: left;  
}
```

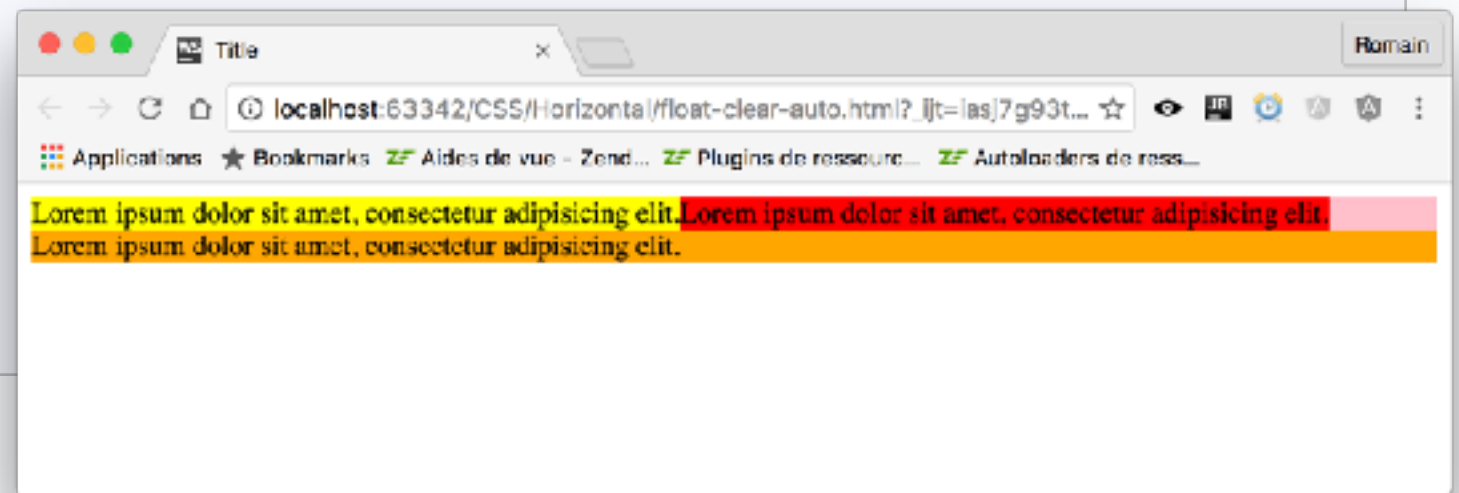


- Un dépassement deviendrait invisible

Positionnement CSS

- Placer 2 éléments block horizontalement

```
.col {  
  float: left;  
}  
  
.container::after {  
  content: "";  
  display: block;  
  clear: left;  
}
```



Positionnement CSS

- Placer 2 éléments block horizontalement

```
.col {  
  display: table-cell;  
}
```



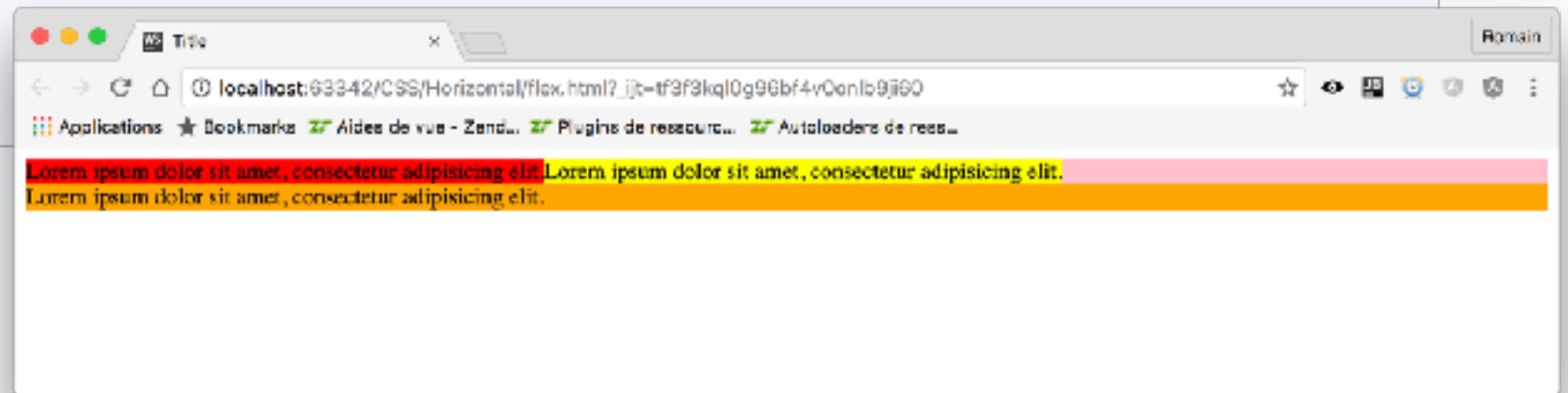
```
.container {  
  display: table;  
  width: 100%;  
}  
  
.col {  
  display: table-cell;  
}  
  
.col.yellow {  
  width: 200px;  
}
```



Positionnement CSS

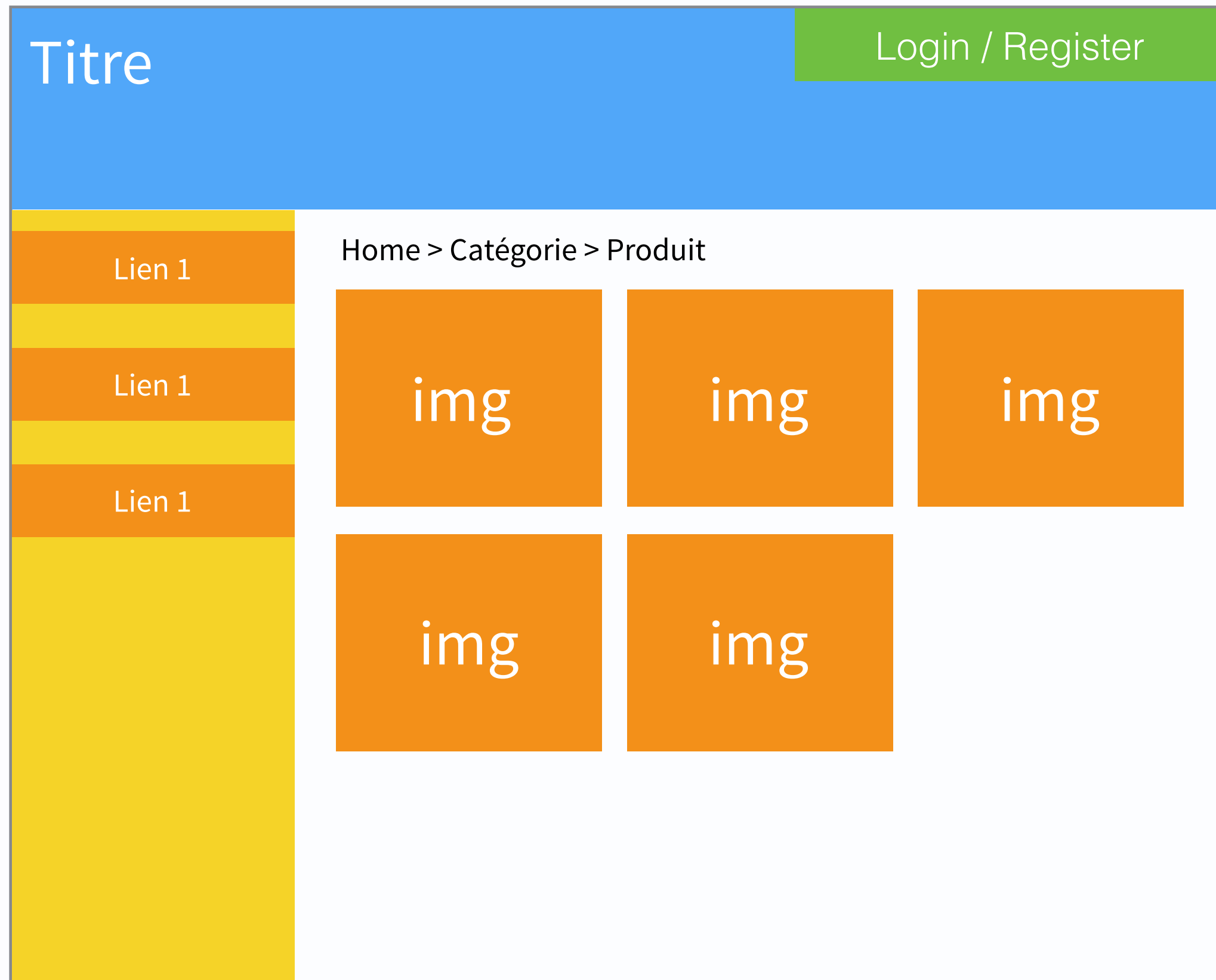
- Placer 2 éléments block horizontalement

```
.container {  
  display: flex;  
}
```

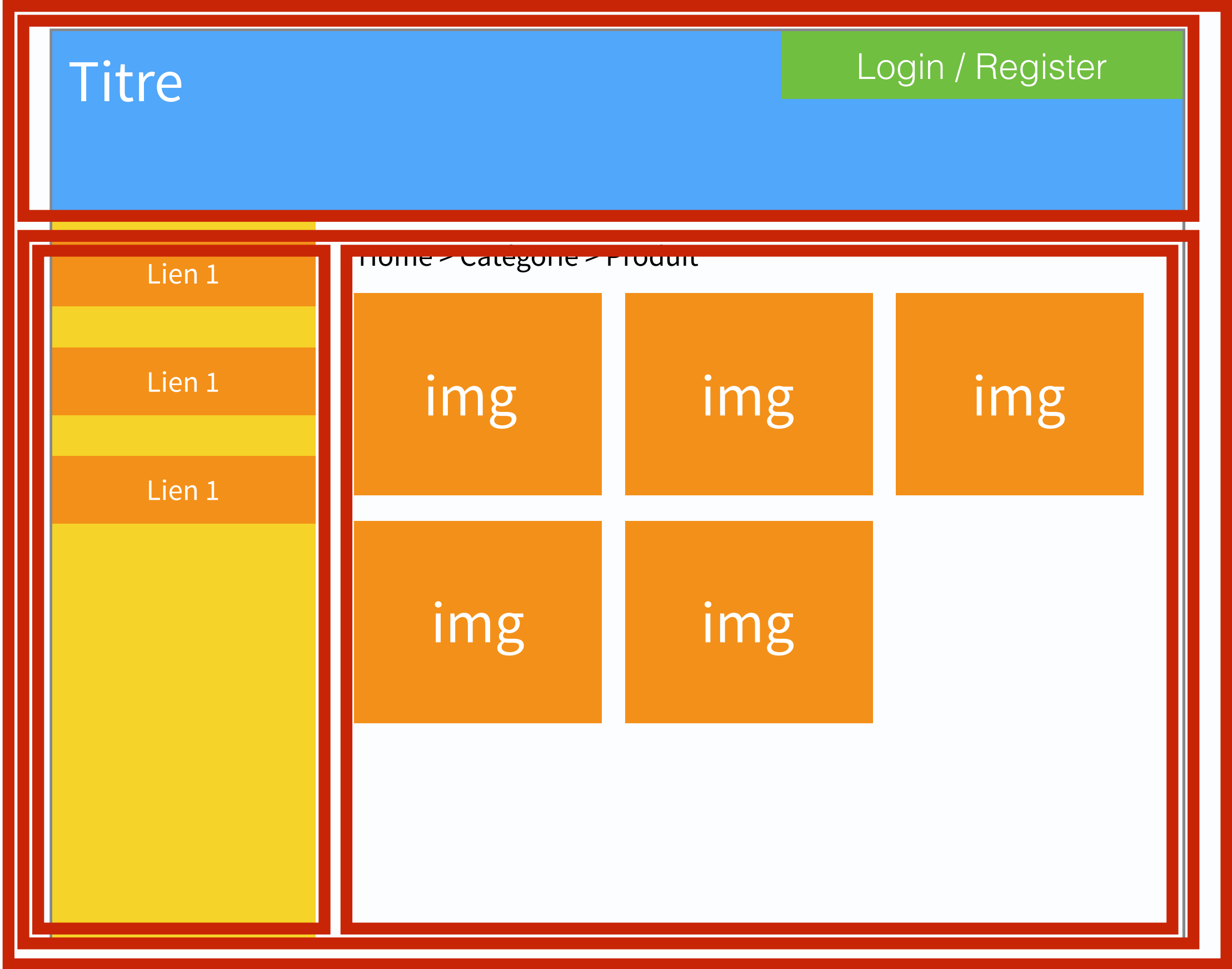


IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			49					4.4	
8	13	47	51			9.2		4.4.4	
11	14	48	52	9.1	39	9.3	all	51	51
		49	53	10	40				
		50	54	TP	41				
		51	55						

Exercice



Exercice



CSS - Flexbox

Flexbox

- Nouveau type de positionnement

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			49					4.4	
8	13	47	51			9.2		4.4.4	
11	14	48	52	9.1	39	9.3	all	51	51
		49	53	10	40				
		50	54	TP	41				
		51	55						

Flexbox

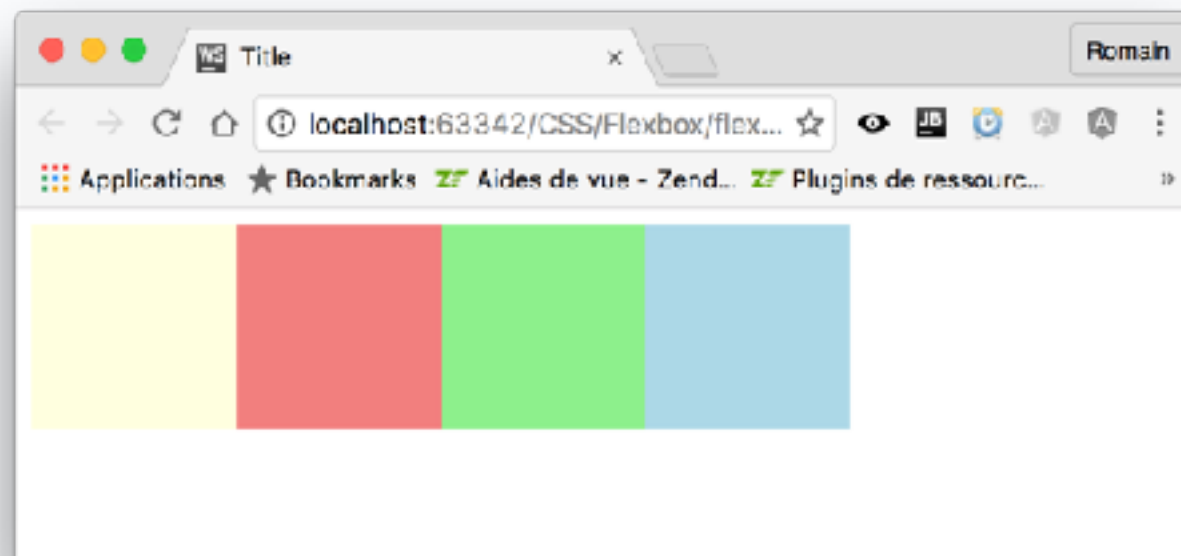
- 4 fonctionnalités principales :
 - Distribution des éléments horizontale ou verticale, avec passage à la ligne autorisé ou non,
 - Alignements et centrages horizontaux et verticaux, justifiés, répartis,
 - Réorganisation des éléments indépendamment de l'ordre du flux (DOM),
 - Gestion des espaces disponibles (fluidité).

Flexbox

► Mise en place

```
.container {  
  display: flex;  
}
```

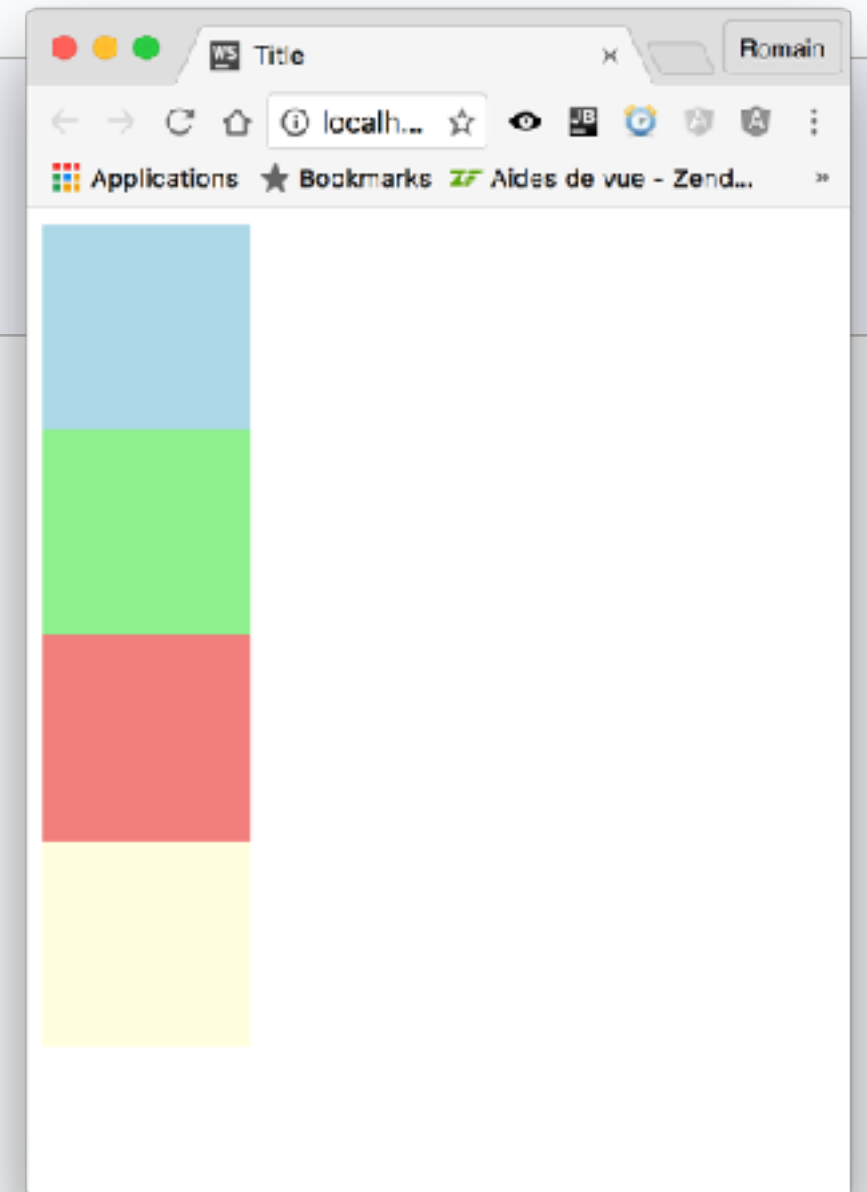
```
<div class="container">  
  <div class="col yellow"></div>  
  <div class="col red"></div>  
  <div class="col green"></div>  
  <div class="col blue"></div>  
</div>
```



Flexbox

► Direction

```
.container {  
  display: flex;  
  flex-direction: column-reverse;  
  /* row column row-reverse column-reverse */  
}
```

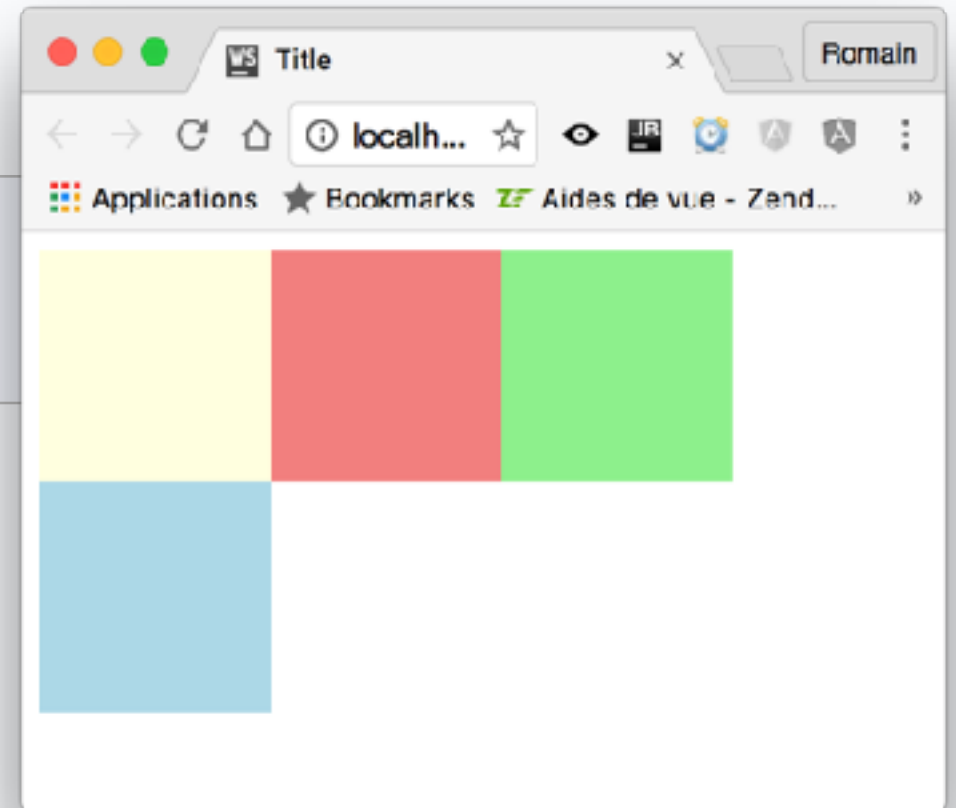


Flexbox

- Passage à la ligne

```
.container {  
  display: flex;  
  flex-wrap: wrap;  
}
```

- Par défaut les éléments ne passent pas à la ligne
- Avec `flex-wrap: wrap-reverse`, la nouvelle ligne sera créée au dessus.



Flexbox

▸ Répartir le contenu

```
.container {  
  display: flex;  
  justify-content: space-around;  
}
```



- flex-start (éléments positionnés au début du sens de lecture, valeur par défaut)
- flex-end (éléments positionnés à la fin)
- center (position centrale)
- space-between (répartition “justifiée”)
- space-around (variante de répartition “justifiée”)

Flexbox

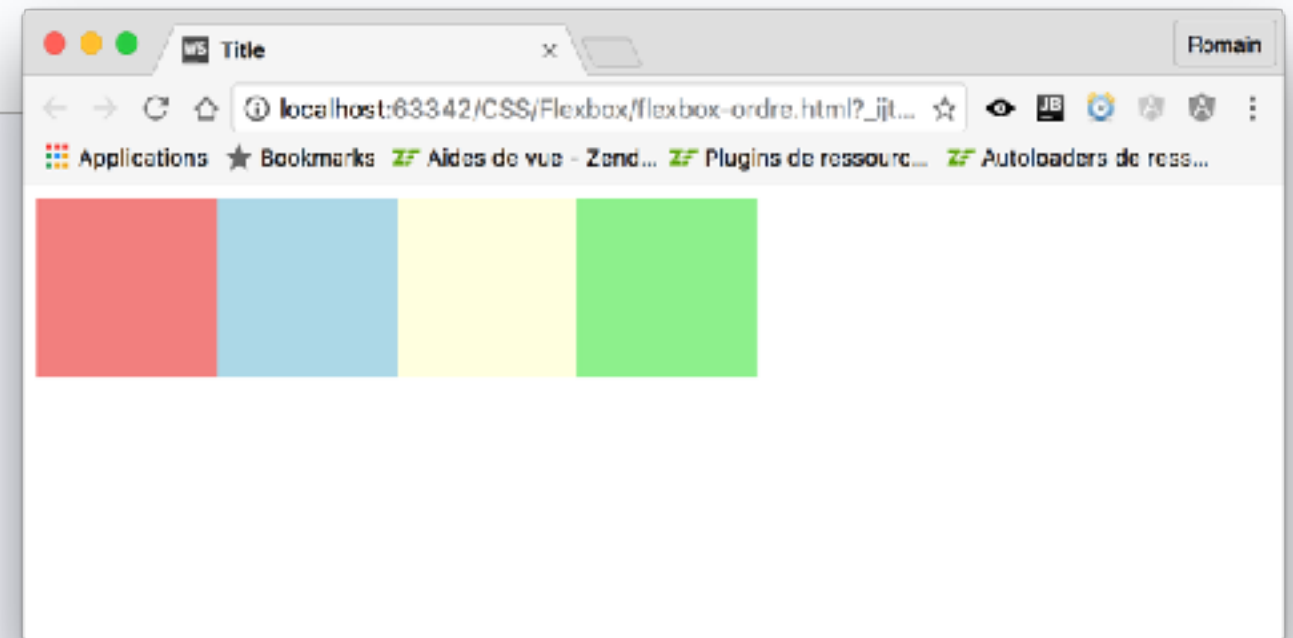
- Axe secondaire: align-items
- Dans l'axe secondaire, les alignements sont régis via la propriété align-items, dont les valeurs sont :
 - flex-start (au début)
 - flex-end (à la fin)
 - center (au centre)
 - baseline (généralement identique à flex-start)
 - stretch (étirés dans l'espace disponible, valeur par défaut)

Flexbox

► Ordonnancement

Il est possible de changer l'ordre. (par défaut 0)

```
.yellow {  
  background-color: lightyellow;  
  order: 3;  
}  
  
.red {  
  background-color: lightcoral;  
  order: 1;  
}  
  
.green {  
  background-color: lightgreen;  
  order: 4;  
}  
  
.blue {  
  background-color: lightblue;  
  order: 2;  
}
```

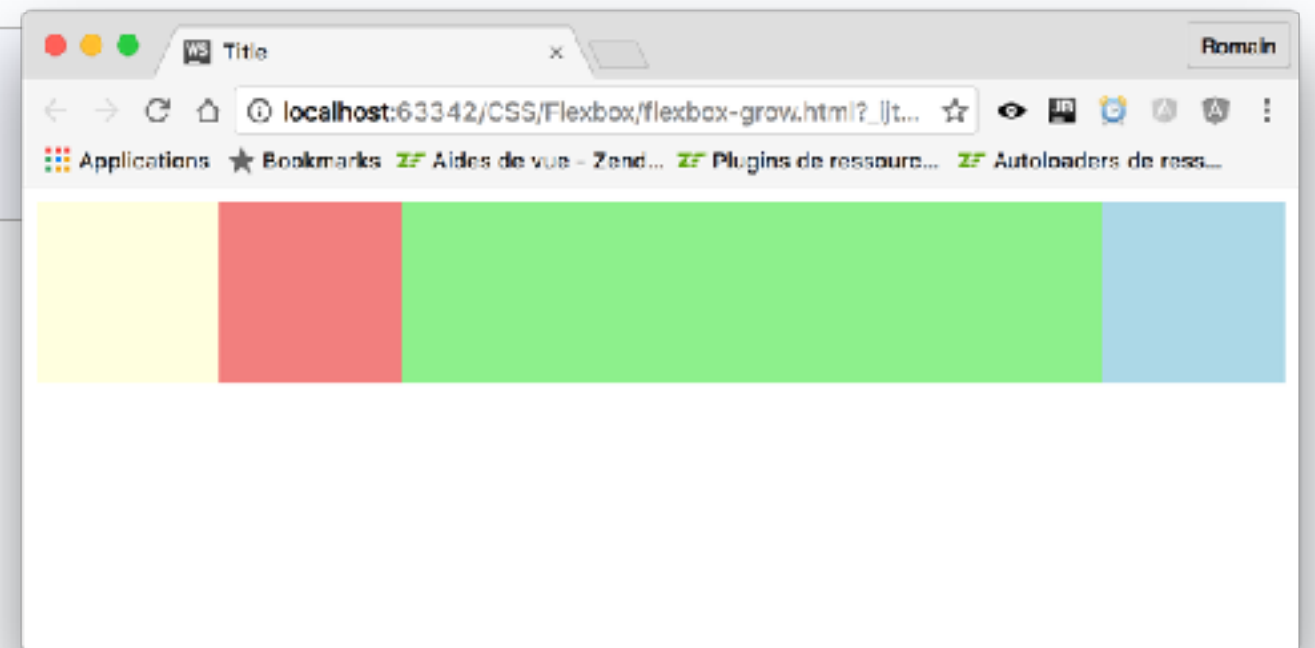


Flexbox

► Flexibilité

Il est possible l'allouer l'espace restant (ici 1/1) à un élément

```
.green {  
  flex: 1;  
}
```



CSS - Web Fonts

Web Fonts

- Possibilité de charger des polices dans le navigateur pour une page web sans les installer sur le système
- Fonctionne dans IE depuis la version 5
- Attention : Licence spéciale sur les polices.
- Temps de chargement à prendre en compte.

Web Fonts

- Google Font Directory : polices libres de droit
<https://fonts.google.com>
- FontSquirrel : Polices + pour préparer ses polices
<http://www.fontsquirrel.com/>
- TypeKit : Polices professionnelles avec licences Web
<http://typekit.com/>

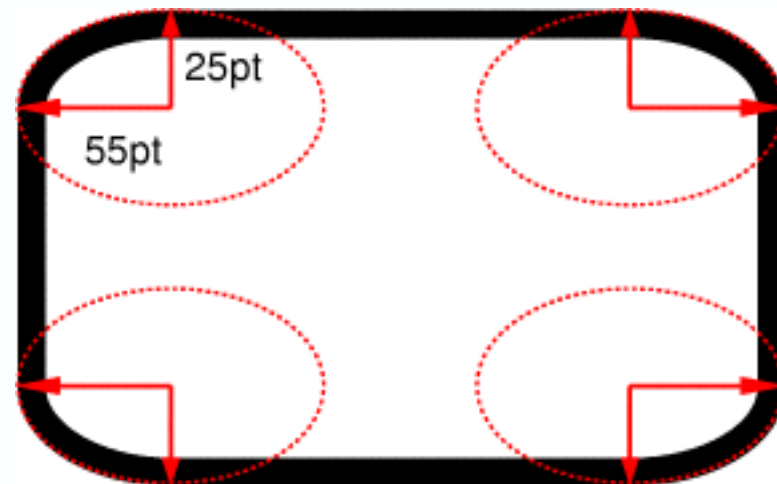
Web Fonts

```
@font-face {  
  font-family: 'CalligraffitiRegular';  
  src: url('Calligraffiti.eot');  
  src: url('Calligraffiti.eot?#iefix') format('embedded-opentype'),  
  url('Calligraffiti.woff') format('woff'),  
  url('Calligraffiti.ttf') format('truetype'),  
  url('Calligraffiti.svg#CalligraffitiRegular') format('svg');  
  font-weight: normal;  
  font-style: normal;  
}  
  
body {  
  font-family: CalligraffitiRegular;  
}
```

CSS - Bords Arrondis

Bords arrondis

- border-top-left-radius: x [y]
- border-top-right-radius
- border-bottom-left-radius
- border-bottom-right-radius
- border-radius: top right bottom left / top right bottom right



CSS - Ombres

Ombres

- `box-shadow: x y blur-x blur-y couleur [inset];`
- On peut en mettre plusieurs séparées par des virgules.
- Elle s'applique de la dernière à la première.
- `inset` = ombre intérieure
- S'adapte aux arrondis.
- `text-shadow: x y blur-radius color|none|initial|inherit;`

CSS - Transformations

Transformations

- transform-origin: centre des transformations précisées dans la propriété transform.
- transform: transformation1 ... transformation(n);
- les transformations sont séparées par un espace.
- Elles sont réalisées de la droite vers la gauche.
- Une matrice de transformation est calculée puis appliquée.

Transformations

- `translate(tx, ty)`, `translateX(tx)`, `translateY(ty)`
- `rotate(angle)` : unité deg, rad ou grad
- `scale(facteur)`, `scale(facteurX, facteurY)`, `scaleX(fx)`, `scaleY(fy)`
- `skew(sx, sy)`, `skewX(sx)`, `skewY(sy)`
- `matrix(m11, m12, m21, m22, tx, ty)` : matrice de transformation 2x2 + translation(tx, ty)

Transformations

- `translate3d(tx, ty, tz)`, `translateX(tx)`, `translateY(ty)`, `translateZ(tz)`
- `rotate3d(ax, ay, az)`, `rotateX(ax)`, `rotateY(ay)`, `rotateZ(az)`
- `scale3d(fx, fy, fz)`, `scaleX(fx)`, `scaleY(fy)`, `scaleZ(fz)`
- `matrix3d(m11, ..., m44)` : matrice de transformation 3D 4x4
- `perspective(profondeur)` : distance du plan $z=0$ par rapport à l'observateur.

CSS - Transitions

Transitions

- Permet d'animer presque tous les changements de valeur des propriétés CSS.
- Etapes :
 - 1. on place la valeur de départ de la propriété + la transition
 - 2. on change la valeur de la propriété à sa valeur d'arrivée.
 - 3. le navigateur applique automatiquement la(les) transition(s)

Transitions

- transition: propriété durée easing délai, ...
- property: none, all, nom d'une propriété
- duration: 0 = pas d'animation, unité : s (secondes) ou ms (millisecondes)
- delay: 0 ou now (au moment du changement de propriété), unité s ou ms.
- timing-function : courbe de timing de l'animation

Transitions

- Basées sur des courbes de bézier :
 - cubic-bezier(point de controle 1, point de controle 2)
 - linear = cubic-bezier(0,0, 1,1)
 - ease = cubic-bezier(0.25, 1, 0.25, 1) - valeur par défaut
 - ease-in = cubic-bezier(0.42, 0, 1, 1)
 - ease-out = cubic-bezier(0, 0, 0.58, 1)
 - ease-in-out = cubic-bezier(0.42, 0, 0.58, 1)

CSS - Animations

Animations

- Définir des animations plus complexes avec des étapes
- On peut animer les mêmes propriétés que pour les transitions.
- Utilisation d'une directive `@keyframes`
- Une série de propriétés :
- `animation-name`, `animation-duration`, `animation-iteration-count`, `animation-delay`, `animation-direction`, `animation-timing-function`

Animations

```
div {  
  width: 100px;  
  height: 100px;  
  background-color: red;  
  animation-name: pulse;  
  animation-delay: 2s;  
  animation-duration: 1s;  
  animation-iteration-count: 10;  
  animation-direction: alternate;  
}  
  
@keyframes pulse {  
  from {  
    transform: scale(1);  
  }  
  30% {  
    transform: scale(0.7);  
  }  
  70% {  
    transform: scale(1.3);  
  }  
  to {  
    transform: scale(1);  
  }  
}
```



npm

npm

- Gestionnaire de dépendance de node.js (s'installe en même temps que node)
- Equivalent pour du code JavaScript à apt-get
- Plutôt destiné à du code console ou serveur, bien que des bibliothèques comme jQuery ou Bootstrap y soient présentes

npm

- Trouver des packages
<https://www.npmjs.com>
- Créer un package
npm init
- Le fichier package.json
<http://browsenpm.org/package.json>

npm

- Installer un package
npm install <package>
npm install <package> --save
npm install <package>@<version> --save

Ex : npm install jquery@1.11.*

- Mettre à jour les packages installés
npm update
- Désinstaller
npm uninstall lodash
npm uninstall --save lodash

Préprocesseurs CSS

Préprocesseurs CSS

- Les préprocesseurs CSS sont des technologies qui permettent à des langages proches de CSS de transpiler en CSS en y ajoutant des fonctionnalités
- Comparsateurs de préprocesseurs CSS
<http://csspre.com/compare/>

Préprocesseurs CSS

- Apparu en 2009, inspiré par SASS
<http://lesscss.org/>
- Variables



```
@link-color:      #428bca; // sea blue

a, .link {
  color: @link-color;
}

.widget {
  color: #fff;
  background: @link-color;
}
```

```
nav ul {
  &:extend(.inline);
  background: blue;
}
```

Préprocesseurs CSS

▸ Mixins (fonctions)

```
.border-radius(@radius) {  
  -webkit-border-radius: @radius;  
  -moz-border-radius: @radius;  
  border-radius: @radius;  
}  
  
#header {  
  .border-radius(4px);  
}  
  
.button {  
  .border-radius(6px);  
}
```

```
.foo {  
  background: #900;  
}  
@import "this-is-valid.less";
```

Préprocesseurs CSS

▸ Imbriquer

```
#header {  
  color: black;  
  .navigation {  
    font-size: 12px;  
  }  
  .logo {  
    width: 300px;  
  }  
}
```

Préprocesseurs CSS

- Transpiler

- Côté client :

```
<link rel="stylesheet/less" type="text/css" href="styles.less" />  
<script src="less.js" type="text/JavaScript"></script>
```

- En ligne de commande

- Installer

```
npm install -g less
```

- Transpiler

```
lessc styles.less > styles.css
```

Préprocesseurs CSS



- Apparu en 2007
<http://sass-lang.com>
- A peu de choses près fonctionnalités égales à Less
- 2 syntaxes :
 - SASS

```
$primary-color: #333  
  
body  
  color: $primary-color
```

- SCSS (inspirée de Less)

```
$primary-color: #333;  
  
body {  
  color: $primary-color;  
}
```

Préprocesseurs CSS

- Apparu en 2010, inspiré par SASS et LESS
<http://learnboost.github.io/stylus/>
- Syntaxe encore plus concise


```
body
  font 12px Helvetica, Arial, sans-serif

a.button
  -webkit-border-radius 5px
  -moz-border-radius 5px
  border-radius 5px
```



Préprocesseurs CSS

- Less et Sass ont inspiré le W3C
Des normes sur les variables, opérateurs existe désormais nativement mais sont mal supportées

calc() as CSS unit value  - CR

Global 76.09% + 5.16% = 81.25%

unprefixed: 75.55% + 5.16% = 80.71%

Method of allowing calculated values for length units, i.e. width:
calc(100% - 3em)

Current aligned Usage relative Show all

IE / Edge	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
8							4.1	
² 9		31					4.3	
10		42					¹ 4.4	
11	38	43	7.1		7.1		¹ 4.4.4	
Edge	39	44	8	30	8.4	8	40	42
	40	45	9	31	9			
	41	46		32				
	42	47						

Préprocesseurs CSS

- Apparu fin 2013
<http://www.myth.io>
- Variables

```
:root {  
  --purple: #847AD1;  
  --large: 10px;  
}  
  
a {  
  color: var(--purple);  
}  
  
pre {  
  padding: var(--large);  
}
```

```
pre {  
  margin: calc(var(--large) * 2);  
}
```


Préprocesseurs CSS



- Apparue mi-2014
<http://cssnext.io>
- Supporte plus de nouveautés CSS que Myth
automatic vendor prefixes, custom properties & ``var()``, reduced ``calc()``, custom media queries, media queries ranges, custom selectors, ``color()``, ``hwb()``, ``gray()``, `#rrggbbaa`, ``rebeccapurple``, font-variant, filter, ``rem`` units, ``:any-link`` pseudo-class, ``:matches`` pseudo-class, ``:not`` pseudo-class, pseudo-elements, Alpha colors, Bonus features, ``@import``, minification, @todo

cssnext {||||}

BEM

BEM

- BEM : Block Element Modifier
- Approche orientée composant
- Méthodologie créé par Yandex en 2010
- Documentation
<https://en.bem.info>

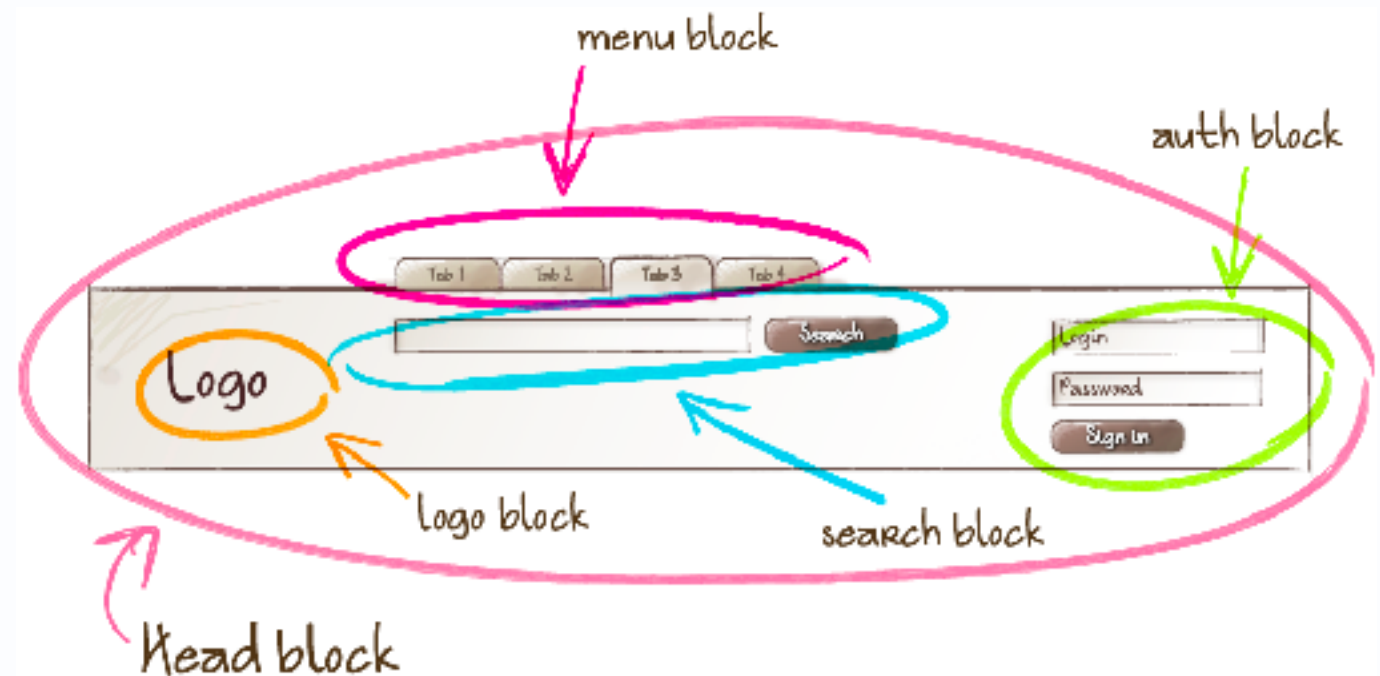
BEM

► Block

Un block est un composant réutilisable sur un site web.

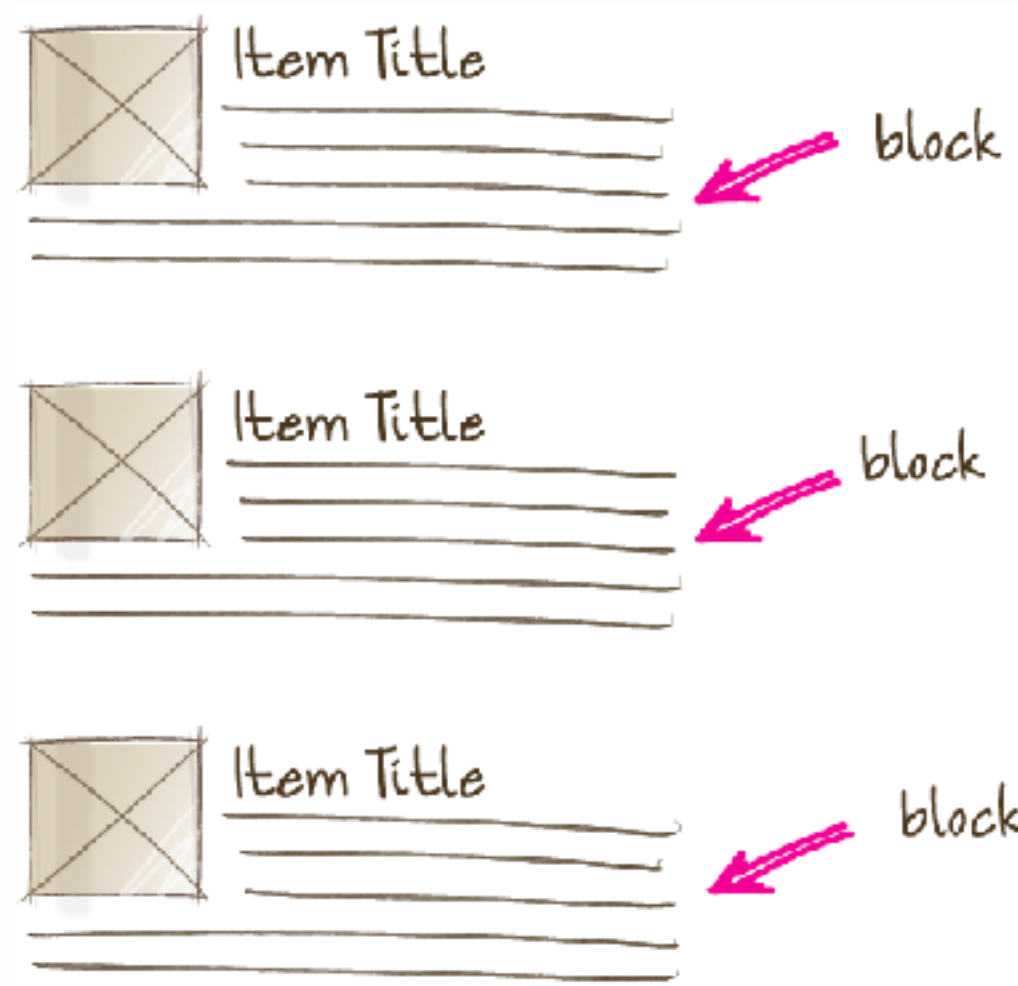
► Exemple :

- Menu
- Bouton
- Barre de recherche
- Formulaire de login
- Footer



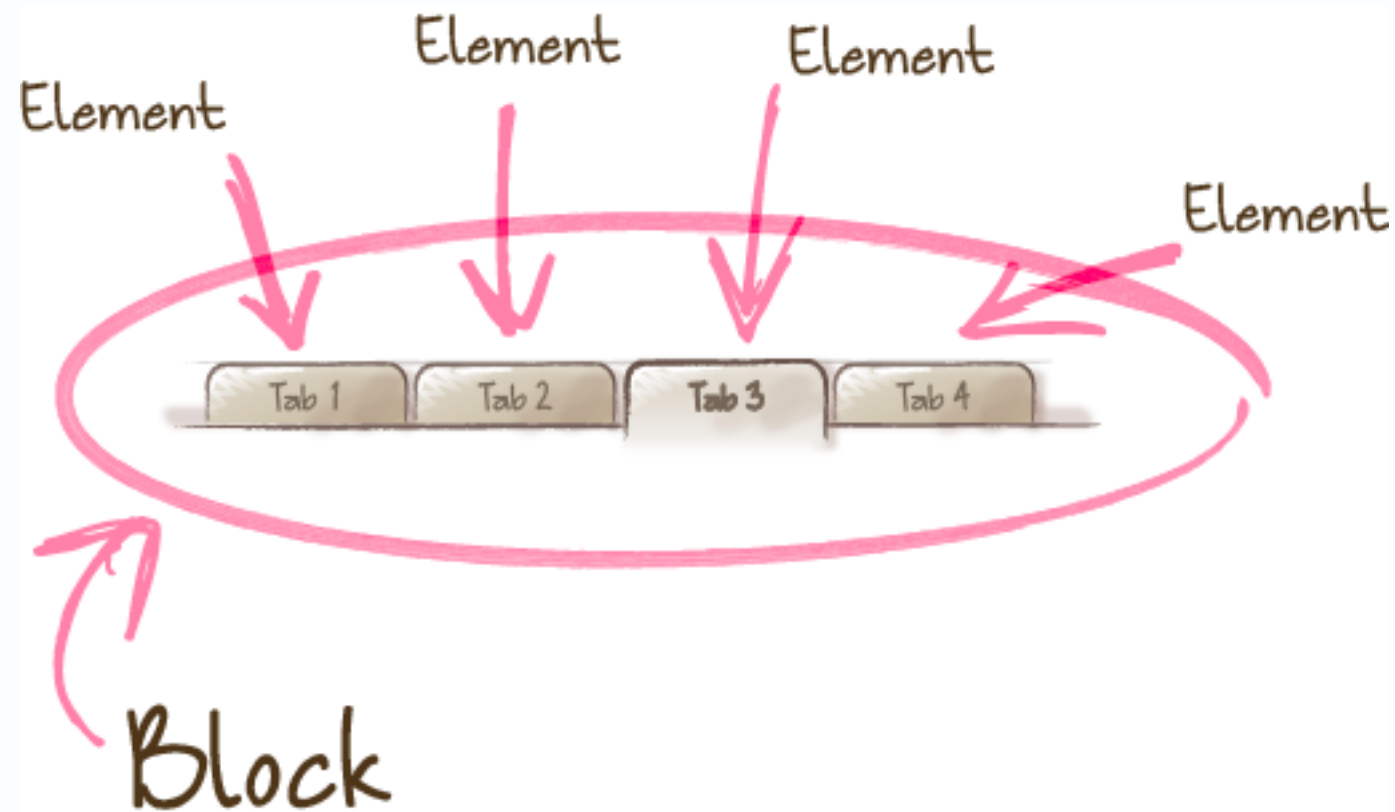
BEM

- Une page doit pouvoir contenir des instances multiples d'un bloc



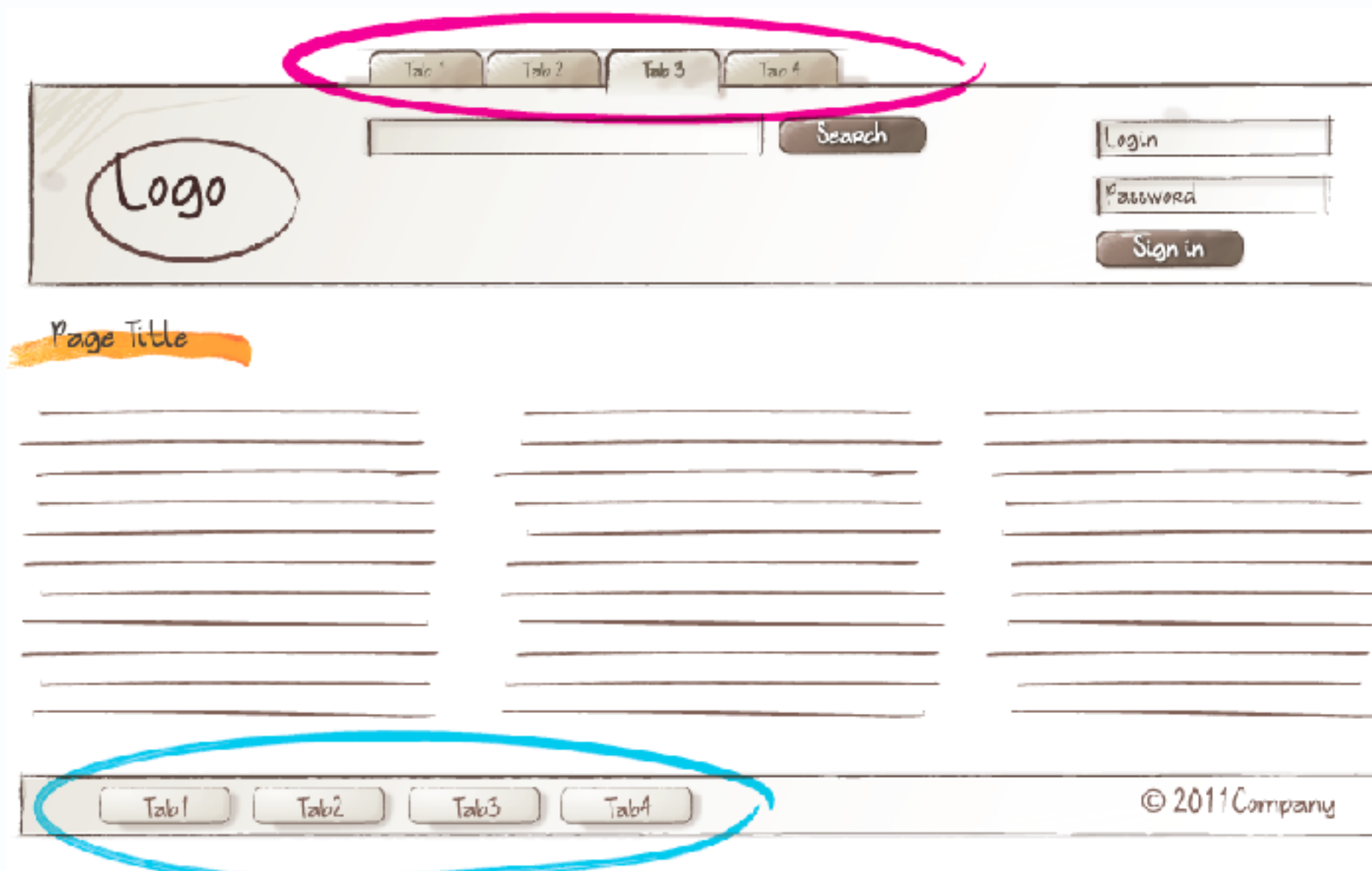
- ▶ Element

Un élément est un composant d'un block qui n'a pas pour vocation à être utilisé en dehors de celui-ci.



► Modifier

Un modifier permet de modifier l'apparence ou le comportement d'un bloc ou d'un élément.



- Conventions de nom

L'approche BEM suggère les conventions de nom suivante.

- Les noms des entités BEM (blocks, elements, modifiers) utilisent des chiffres et lettres non accentuées.
- Les mots sont séparés par des traits d'union (-).
- Les noms utilisent des classes CSS

- Pour les Entités :

- Blocks (parfois CamelCase), exemples : menu, form-login
- Elements, reprend le nom du block suivi d'un double underscore (__), exemple : menu__item, form-login__input-email
- Modifiers, reprend le nom du block suivi d'un simple underscore (_) (parfois double trait d'union (--)), exemple : menu_active, selecteur-langue_theme_windows

► Organisation des fichiers

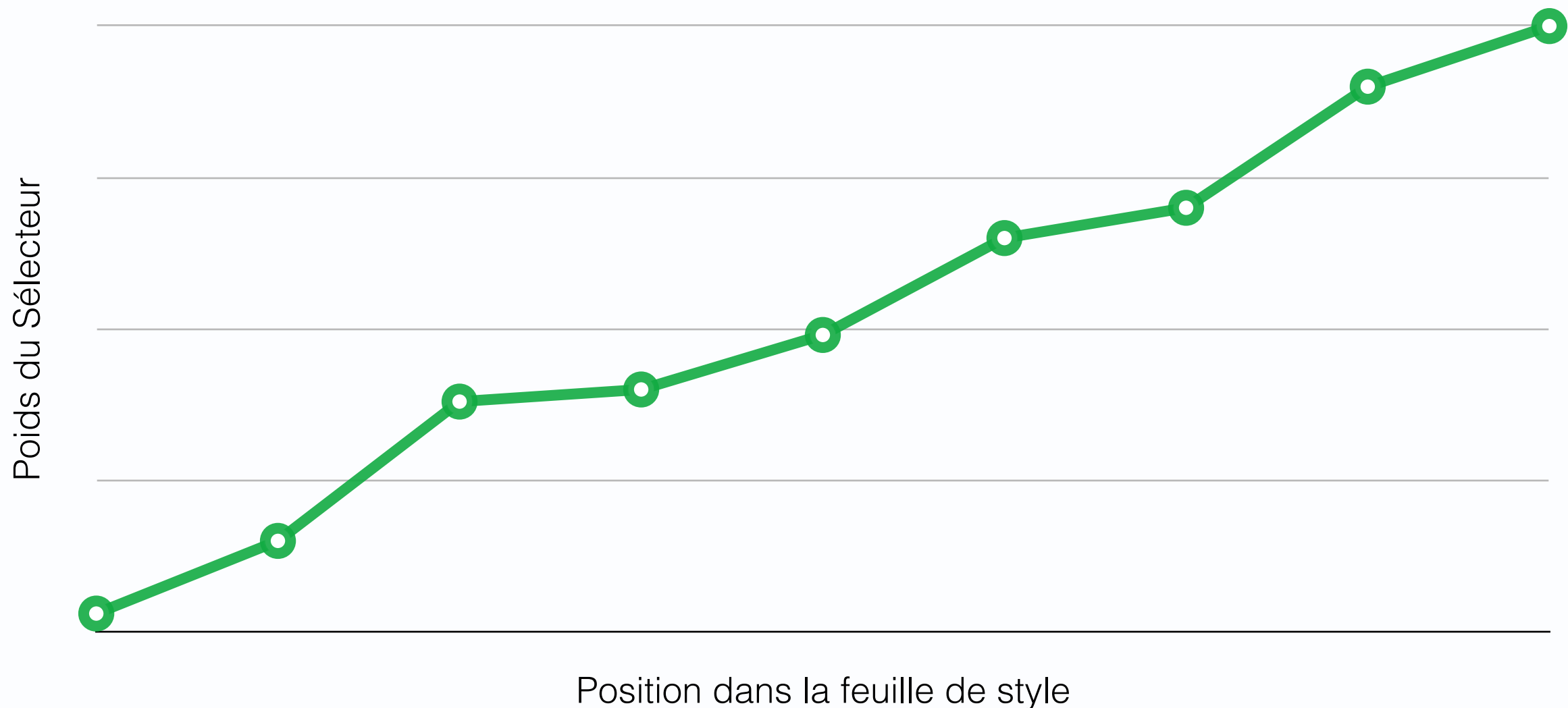
```
blocks/  
  input/  
    _type/  
      input_type_search.css  
    __box/  
      input__box.css  
    __clear/  
      _visible/  
        input__clear_visible.css  
      _size/  
        input__clear_size_large.css  
      input__clear.css  
      input__clear.js  
    input.css  
    input.js  
  button/  
    button.css  
    button.js  
    button.png
```

*# type modifier directory
Implementation of modifier type
with value search in CSS technology
box element directory

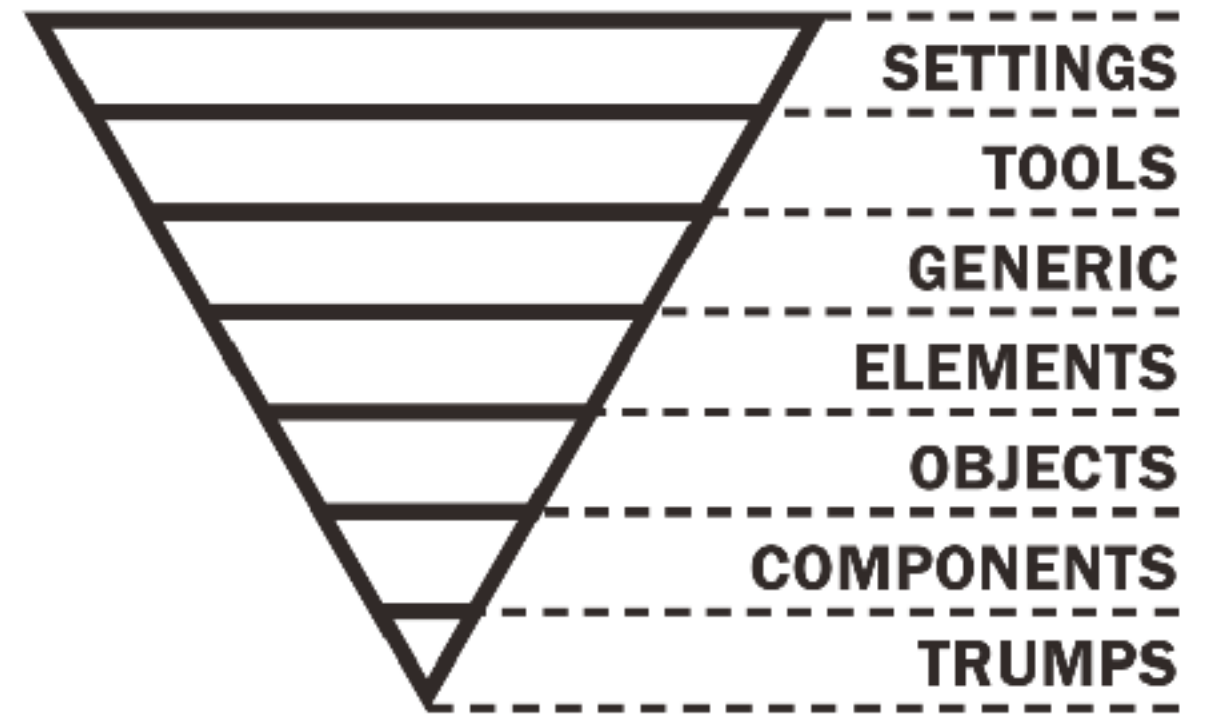
clear element directory
visible modifier directory
Implementation of boolean modifier visible
with value true in CSS technology
size modifier directory
Implementation of modifier size
with value large in CSS technology*

ITCSS

- Organisation du CSS par couche
- Ecriture de la priorité de sélecteur la plus faible à plus forte



- Settings : variables
- Tools : mixins
- Generic : Resets, Normalize.css
- Elements : style de base, h1-h6, body, lists
- Objects : composants génériques .ui-list
- Components : composants spécialisés .products-list
- Trumps : écrasement de propriétés, !important



Certaines couches peuvent être ajoutées (themes...) ou retirés (pas de Préprocesseur = pas de settings ou tools)

ITCSS

```
@import "settings.colors";
@import "settings.global";

@import "tools.mixins";

@import "normalize-scss/normalize.scss";
@import "generic.reset";
@import "generic.box-sizing";
@import "generic.shared";

@import "elements.headings";
@import "elements.forms";
@import "elements.links";
@import "elements.quotes";
@import "elements.tables";

@import "objects.animations";
@import "objects.drawer";
@import "objects.layout";
@import "objects.overlays";

@import "components.404";
@import "components.about";
@import "components.archive";
@import "components.avatars";
@import "components.blog-post";
@import "components.topbar";
@import "components.work-list";
@import "components.work-detail";

@import "trumps.clearfix";
@import "trumps.utilities";
```