



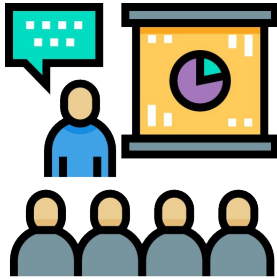
# Développement Front-End (partie 1)

**Aous Karoui**

[aous.karoui@iut2.univ-grenoble-alpes.fr](mailto:aous.karoui@iut2.univ-grenoble-alpes.fr)

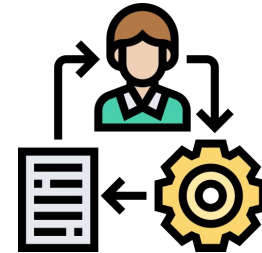


# Organisation



- **Volume horaire** : 30h
- **Répartition** : 7 x 4h + 2h (examen pratique)
- Cours / TD (projet file rouge)

- **Évaluation** : Examen pratique + projet fil rouge



# Organisation



- **Partie 1 : prog.statique**
- **HTML 5**
- **CSS 3**
  - Flexbox
  - Sass
  - Compass



- **Partie 2 : prog.dynamique**
- Javascript
- jQuery
- Js orienté objet
- AJAX



- **Partie 3 : web responsive**
- Media queries (css)
- Bootstrap
- + Canvas





# Partie 1

## Programmation statique

# HTML 5



- 1. Présentation
- 2. Structure
- 3. Typologie des éléments
- 4. TP1



## Mieux vous connaître...

---

QUI d'entre vous ?

- A. Connaît bien / très bien HTML5
- B. Connaît un peu HTML5 / un autre HTML (HTML4, XHTML...)
- C. Ne connaît absolument pas HTML5

--	--	--

A

B

C



- 1. Présentation
- 2. Structure
- 3. Typologie des éléments
- 4. TP

# 1. Présentation



## Historique

- HTML : **H**yper**T**ext **M**arkup **L**anguage
  - documents hypertextuels avec hyperliens
- Une application du langage SGML
- Créé en 1992 par Tim Berners-Lee



Tim Berners-Lee (1990)

## Chronologie

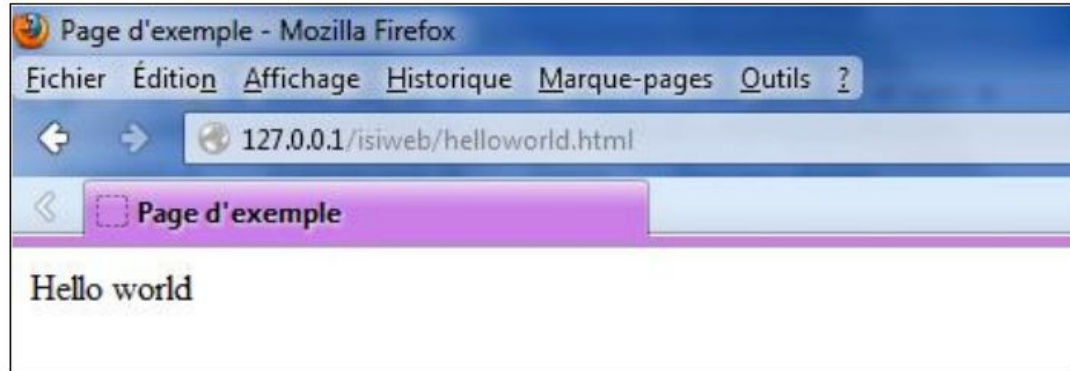
- GML (IBM 1980)
- SGML (Charles Goldfarb 1986)
- HTML (Tim Berners-Lee 1992)
- HTML 2.0 (1995-1996)
- HTML 4 (1997)
- XHTML (2000)
- HTML 5 (2007)

# 1. Présentation

## Structure d'un fichier HTML



```
<html>
  <head>
    <meta charset="ISO-8859-1">
    <title>Page d'exemple</title>
  </head>
  <body>
    <p>Hello world</p>
  </body>
</html>
```





# 1. Présentation

## Structure d'un fichier HTML



```
<html>  
  <head>  
    <meta charset="ISO-8859-1"/>  
    <title>Page d'exemple</title>  
  </head>  
  <body>  
    <p>Hello world</p>  
  </body>  
</html>
```

1- Des codes entre  
< et > : les **balises** !

# 1. Présentation

## Structure d'un fichier HTML



```
<html>
  <head>
    <meta charset="UTF-8">
    <title>Page d'exemple</title>
  </head>
  <body>
    <p>Hello world</p>
  </body>
</html>
```

2- Du contenu

# 1. Présentation



## Nouveautés dans HTML 5

- Déclarations simplifiées

### XHTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

### HTML4

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

### HTML5

```
<!DOCTYPE html>
```

# 1. Présentation



## Nouveautés dans HTML 5

- Déclarations simplifiées
- Nouvelles balises sémantiques `<header>`, `<footer>`, `<article>`, `<section>`
- Contrôles de saisie `number`, `date`, `time`, `calendar`, `range`
- Intégration audio, vidéo `<audio>`, `<video>`
- Intégration d'éléments graphiques `<svg>`, `<canvas>`  
(a remplacé l'usage de flash, silverlight...)
- ...



- 1. Présentation
- 2. Structure
- 3. Typologie des éléments
- 4. TP

## 2. Structure



```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Page d'exemple</title>
  </head>
  <body>
    <p>Hello world</p>
  </body>
</html>
```

2- Des blocs : le document  
**Html** qui contient un **en-**  
**tête** et un **corps**

## 2. Structure

### Déclaration



```
<!doctype html>
```

- Le tag doctype indique au navigateur dans quel type de HTML la page a été écrite
  - ([HTML-3.2 «classique», HTML-4 de transition ou strict, XHTML, etc...](#))
  - <!doctype html> signifie : document en HTML5
- Utile pour les navigateurs, les lecteurs de votre code ou les validateurs de code

## 2. Structure

### Langue



```
<html lang="fr">
```

- La déclaration de la langue se fait dans la balise html  
`<html lang="fr">`
- Utile pour le référencement et la synthèse vocale
  - (ex : Google vous propose les pages en français ou une traduction)



## 2. Structure

### En-tête



```
<head>  
  <meta charset="UTF-8">  
  <title>Page d'exemple</title>  
</head>
```

- Délimitée par les balises `<head>` et `</head>`
- Contient des informations à titre indicatif
  - **Le titre**
  - **Le type d'encodage des caractères**
  - **Les méta-données**
  - **Les références à d'autres ressources**

# 2. Structure

## En-tête



- Les méta-données

- mots clés et description
- informations / instructions aux moteurs de recherche
- ex: `<metaname="description" content="Cours LP, IUT2 Grenoble"/>`  
`<metaname="author" content="Pierre Dupont">`  
`<meta name="robots" content="[[no]index][no[follow]]"/>`

- Les références

- Référencer des ressources utilisées par le document: feuilles de style CSS, icône, fichiers de scripts externes Javascript
- ex: `<link rel="stylesheet" type="text/css" href="../main.css">`  
`<link rel="shortcut icon" type="image/x-icon" href="favicon.ico" />`

## 2. Structure

### Corps

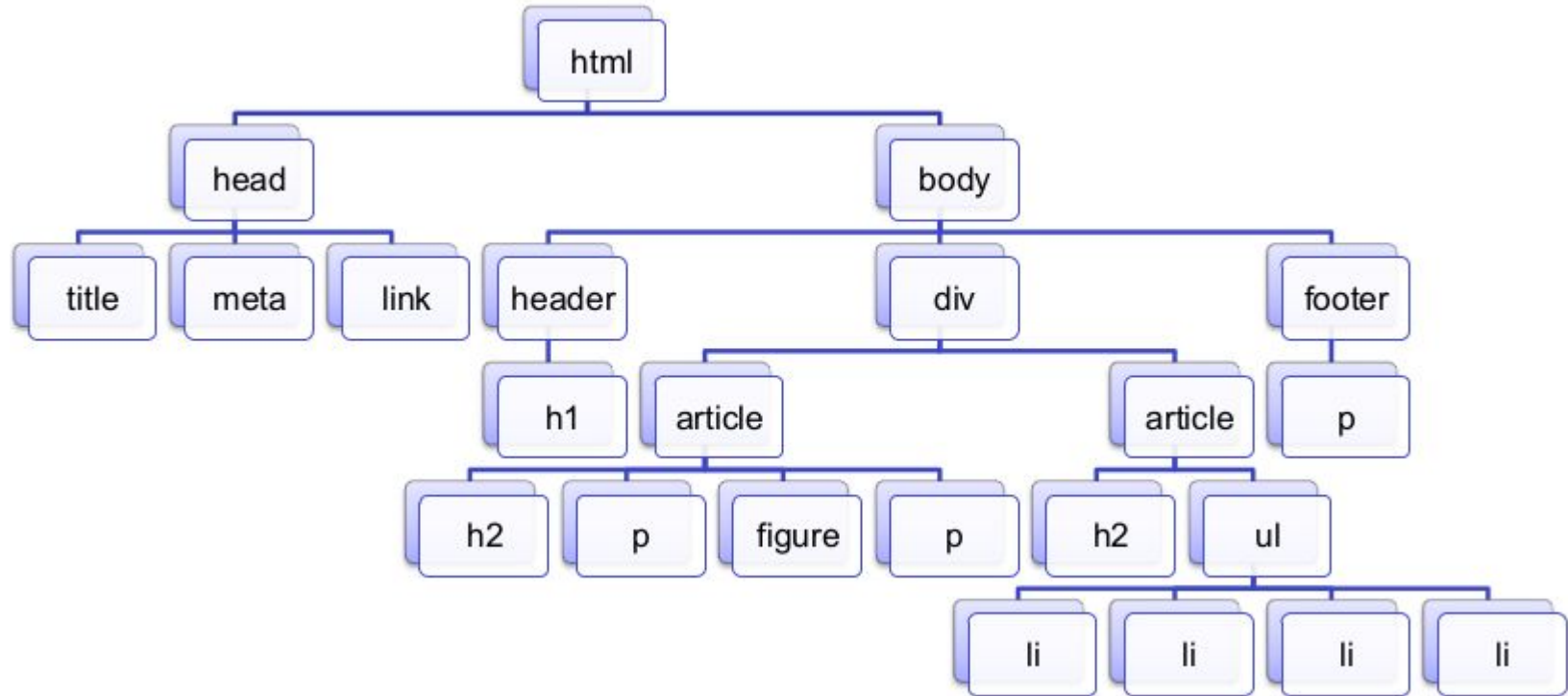


```
<body>  
    <p>Hello world</p>  
</body>  
</html>
```

- Délimitée par les balises `<body>` et `</body>`
- Peut contenir les éléments suivants
  - du **texte** (titres, paragraphes, listes, etc.)
  - des **images**
  - des **hyperliens**
  - des **tableaux**
  - des **formulaires**
  - des **cadres**
  - des objets externes (applets Java, Flash ...)
  - etc.

## 2. Structure

### Arbre d'une page HTML (exemple)





- 1. Présentation
- 2. Structure
- 3. Typologie des éléments
- 4. TP



# 3. Typologie des éléments

## Balises de type block

- Des blocs dans les documents
  - Exemples : paragraphes, listes...
- Apparaissent les uns en dessous des autres
- Ont des dimensions et des marges fixées par défaut (à l'exception des blocs `<div>`)
- Sont positionnables (avec les feuilles de style CSS)
- Peuvent contenir d'autres blocs (sauf les blocs de paragraphes `<p>` et de titres `<h1>`, `<h2>`, ... `<h6>`)
- Peuvent contenir des éléments inline



# 3. Typologie des éléments

## Balises de type block

- `<h1></h1>`
- `<h2></h2>`
- ...
- `<h6></h6>`
- `<p></p>`
- `<table></table>`
- `<ul></ul>`
- `<ol></ol>`
- `<blockquote></blockquote>`
- `<dl></dl>`
- `<div></div>`
- etc.



# 3. Typologie des éléments

## Balises de type inline

- Apparaissent au fil du texte
- Servent à modifier, enrichir des portions de textes, apporter du sens
- Ne peuvent contenir que des éléments inline(→ pas de block)
- Un élément inline doit être contenu dans un élément de type block





# 3. Typologie des éléments

## Balises de type inline

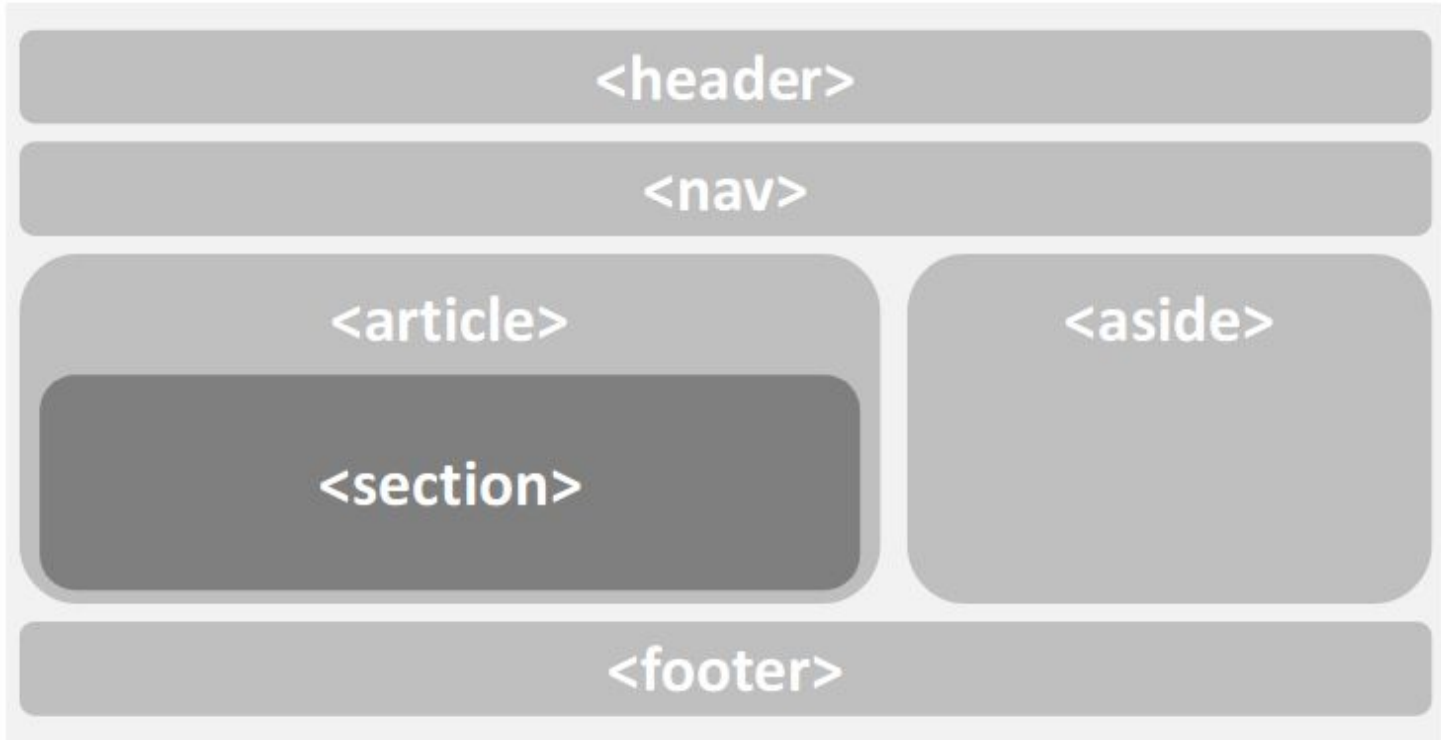
Exemples :

- `<a></a>`
- `<em></em>`
- `<img>`
- `<b></b>`
- `<i></i>`
- `<strong></strong>`
- `<span></span>`



# 3. Typologie des éléments

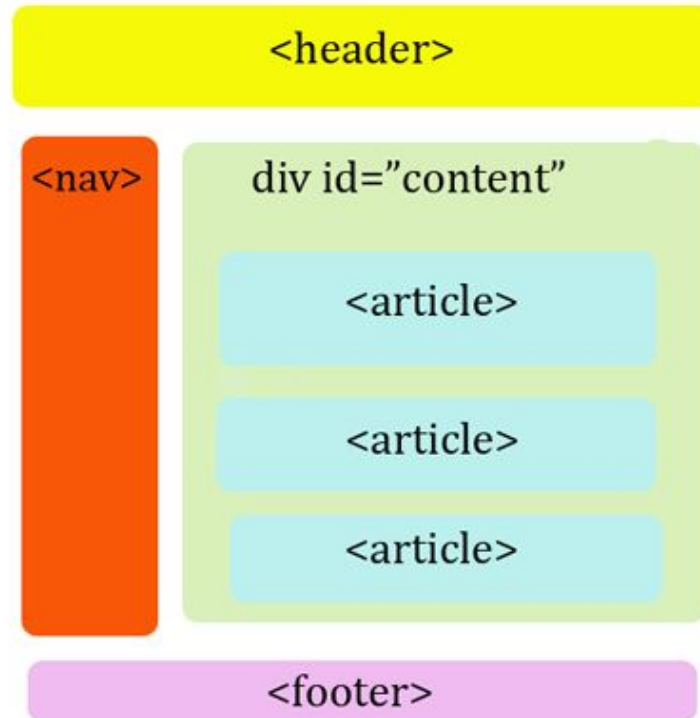
## Balises sémantiques





# 3. Typologie des éléments

## Balises sémantiques



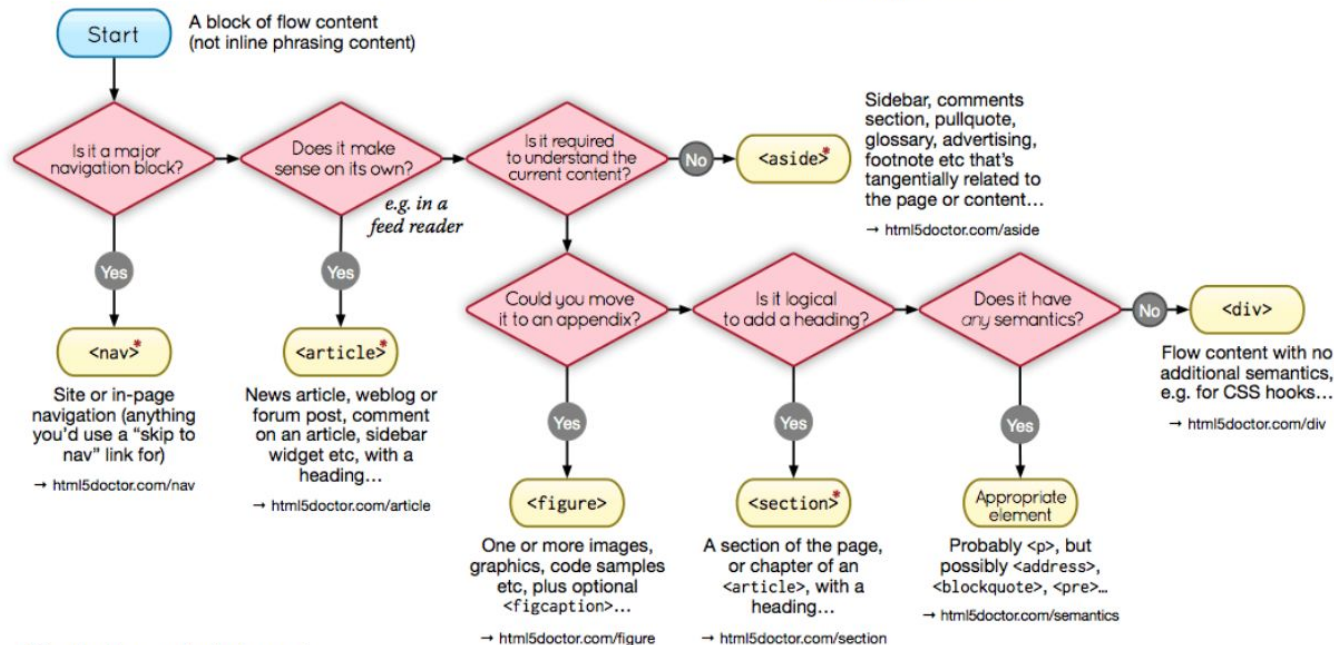
# 3. Typologie des éléments

## Balises sémantiques



### HTML5 Element Flowchart Sectioning content elements and friends

By @riddle & @boblet  
[www.html5doctor.com](http://www.html5doctor.com)



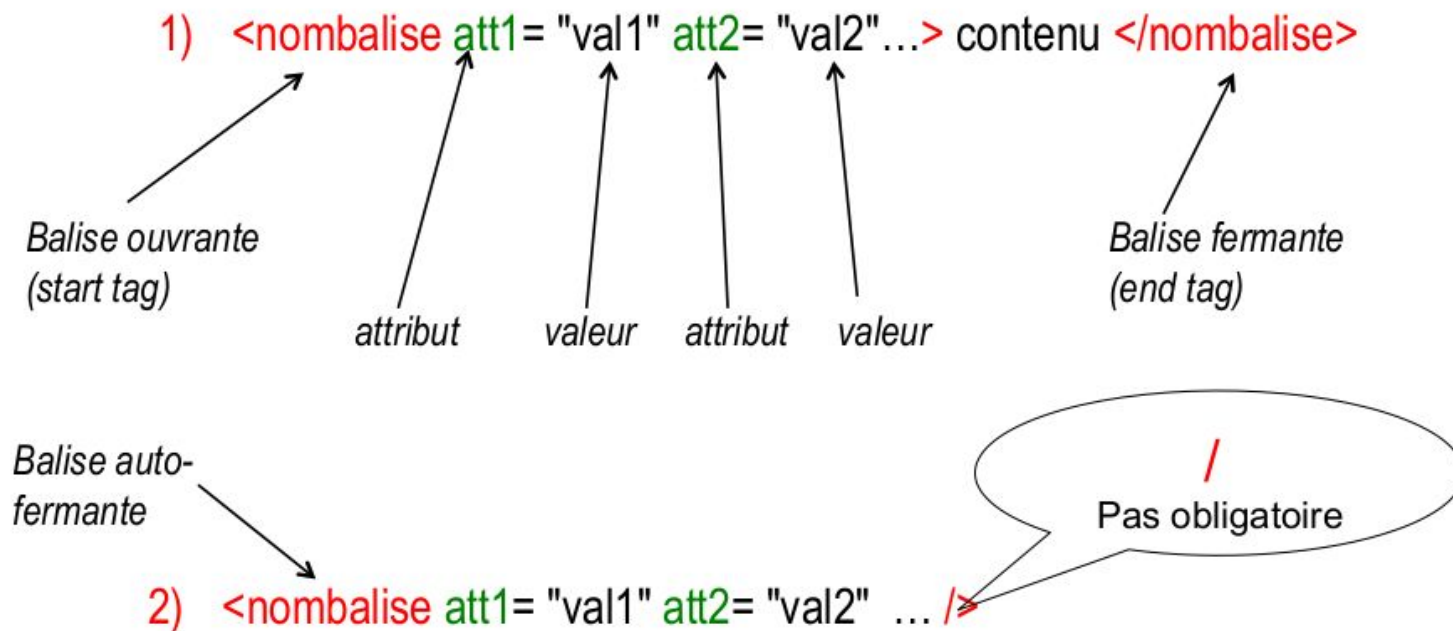
\* Sectioning content element

These four elements (and their headings) are used by HTML5's outlining algorithm to make the document's outline  
→ [html5doctor.com/outline](http://html5doctor.com/outline)

2011-07-22 v1.5  
For more information:  
[www.html5doctor.com/semantics](http://www.html5doctor.com/semantics)

## 2. Structure

### Forme générale des éléments





- 1. Présentation
- 2. Structure
- 3. Typologie des éléments
- 4. TP

# 4. TP1



Avant de démarrer votre 1er TP, assurez-vous de choisir l'éditeur qui vous convient

Éditeur	Licence	Prix	Système d'exploitation	Support	Documentation	Extensible
<a href="#">Atom</a>	MIT/BSD	Gratuit	Windows, Mac, Linux	<a href="#">Forum</a>	<a href="#">Manuel en ligne</a>	<a href="#">Oui</a>
<a href="#">Brackets</a>	MIT/BSD	Gratuit	Windows, Mac, Linux	<a href="#">Forum</a> , <a href="#">IRC</a>	<a href="#">Wiki GitHub</a>	<a href="#">Oui</a>
<a href="#">Coda</a>	Propriétaire	99 \$	Mac	<a href="#">Twitter</a> , <a href="#">Forum</a> , e-mail	<a href="#">eBook</a>	<a href="#">Oui</a>
<a href="#">Emacs</a>	GPL 3	Gratuit	Windows, Mac, Linux	<a href="#">FAQ</a> , <a href="#">liste de diffusion (anglophone)</a> , News Group	<a href="#">Manuel en ligne</a> / Tutoriel en français depuis l'éditeur / <a href="#">pages sur Ubuntu-fr</a>	Oui
<a href="#">Espresso</a>	Propriétaire	75 \$	Mac	<a href="#">FAQ</a> , e-mail	Pas de documentation utilisateur mais il existe une <a href="#">documentation pour les extensions (plug-ins)</a>	Oui
<a href="#">Gedit</a>	GPL	Gratuit	Windows, Mac, Linux	<a href="#">Liste de diffusion (anglophone)</a> , IRC	<a href="#">Manuel en ligne (en français)</a> / <a href="#">Introduction sur Ubuntu-fr</a>	<a href="#">Oui</a>
<a href="#">Komodo Edit</a>	MPL	Gratuit	Windows, Mac, Linux	<a href="#">Forum</a>	<a href="#">Manuel en ligne</a>	<a href="#">Oui</a>
<a href="#">Notepad++</a>	<a href="#">GPL modifiée</a>	Gratuit	Windows	<a href="#">Forum</a>	<a href="#">Wiki</a>	<a href="#">Oui</a>
<a href="#">PSPad</a>	Propriétaire	Gratuit	Windows	<a href="#">FAQ</a> , <a href="#">Forum</a>	<a href="#">Aide en ligne</a>	<a href="#">Yes</a>
<a href="#">Sublime Text</a>	Propriétaire	\$70	Windows, Mac, Linux	<a href="#">Forum</a>	<a href="#">Documentation officielle</a> , <a href="#">documentation officieuse</a>	<a href="#">Oui</a>

# 4. TP1



- Des outils utiles pour vérifier / valider son code HTML
  - <https://validator.w3.org/>
  - <https://www.freeformatter.com/html-validator.html>

Attention : Validation uniquement syntaxique (non sémantique)

**TP 1 sur Chamilo**





# Partie 1

## Programmation statique (suite)



- 1. Présentation
- 2. Les règles d'écriture
- 3. Sélecteurs simples
- 4. Sélecteurs avancés
- 5. TP2

# 1. Présentation



## Historique

- ❖ CSS : **C**ascading **S**tyle **S**heets
  - Feuilles de style en cascade
- ❖ Créé en 1995 par [Håkon Wium Lie](#) et [Bert Bos](#)



- CSS (1995)
- CSS 2 (1997)
- CSS 2.1 (2001)
- CSS 3 (2007)

## Objectif



- ❖ Séparer la structure et la présentation

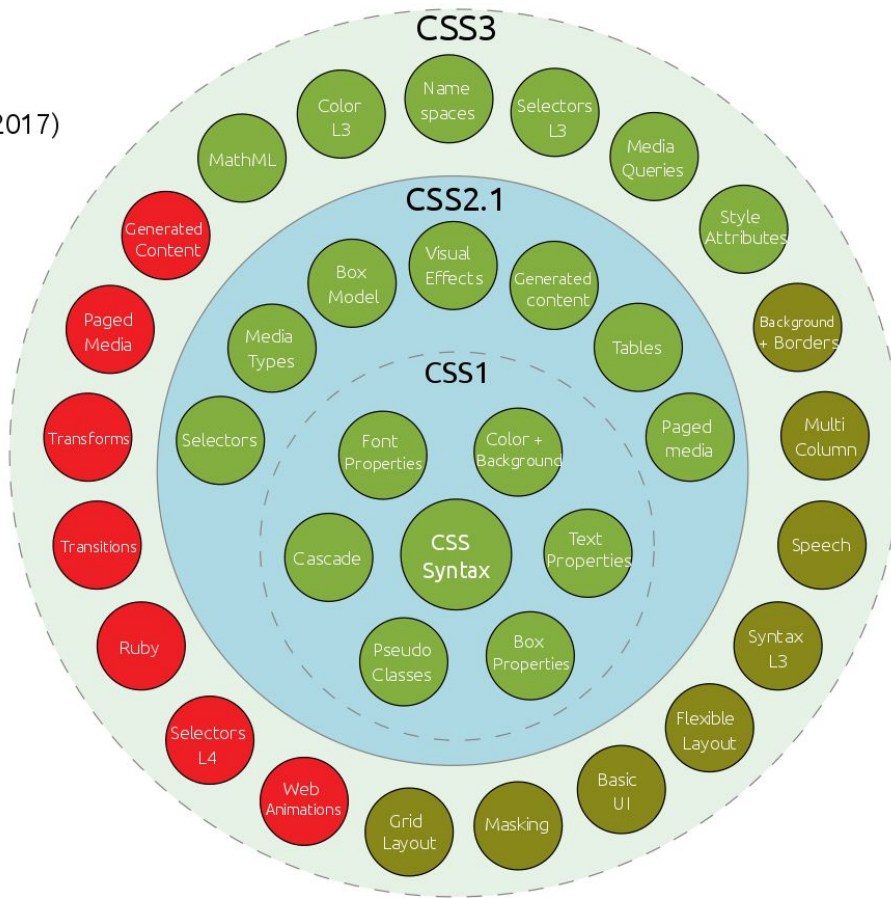
# 1. Présentation



## CSS3

Taxonomy & Status (September 2017)

-  W3C Recommendation
-  Candidate Recommendation
-  Last Call
-  Working Draft
-  Obsolete or inactive



# 1. Présentation



## Une page web (codage HTML5 + CSS3) document numérique *hypertexte*

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Un exemple </title>
    <link rel="stylesheet" type="text/css" href="style.css" />
  </head>
  <body>
    <header>
      <h1>Les bons plans étudiants pour un jeudi après-midi sportif !</h1>
    </header>
    <div>
      <article>
        <h2>Via Ferrata</h2>
        <p>
          La via ferrata de la Bastille est située en plein Grenoble, ce qui est très pratique.
        </p>
        
        <p>
          Mais il y a aussi celle de Crolles qui n'est qu'à quelques kilomètres.
        </p>
      </article>
      <article>
        <h2>Stations de ski proches de Grenoble</h2>
        <ul>
          <li>Chamrousse</li>
          <li>Les Sept Laux</li>
          <li>Villard de Lans</li>
          <li>Autrans</li>
        </ul>
      </article>
    </div>
    <footer>
      <p>Copyright &copy; 2013 &mdash; IUT2 Grenoble &mdash; Cours M115</p>
    </footer>
  </body>
</html>
```

```
/* -- body -- */
body {
  width: 60%;
  margin: auto;
  font-family: georgia, serif;
  font-size: 1em;
  color: #666;
}

/* -1- header -- */
header {
  margin: 10px 0 10px 0;
  text-align: center;
  font-variant: small-caps;
}

/* -2- div -- */
div {
  margin: 0 0 5px 0;
}

article {
  padding: 5px 5px 5px 5px;
  margin: 0 0 5px 0;
  border: 1px #666 solid;
}

article h2 {
  color: hsla(240, 100%, 50%, 0.5);
  text-align: center;
}

/* -3- footer -- */
footer {
  margin: 0 0 0 0;
  text-align: center;
}
```

Fichiers page.html et style.css

### LES BONNS PLANS ÉTUDIANTS POUR UN JEUDI APRÈS-MIDI SPORTIF !

#### Via Ferrata

La via ferrata de la Bastille est située en plein Grenoble, ce qui est très pratique.



Mais il y a aussi celle de Crolles qui n'est qu'à quelques kilomètres.

#### Stations de ski proches de Grenoble

- Chamrousse
- Les Sept Laux
- Villard de Lans
- Autrans

Copyright © 2013 - IUT2 Grenoble - Cours M115

Résultat: les fichiers ouverts avec un navigateur comme chrome, firefox,...

# 1. Présentation



## Principe

- Sélectionner des éléments HTML
- Leur appliquer du style

## Apports

- **Séparation** du contenu et de la présentation•
- **Positionnement** et style
- Construction d'une charte graphique(et d'une cohérence globale)
- **Factorisation** des règles de style

## 2. Les règles d'écriture



- ❖ Une **règle** se compose de :
  - Un **sélecteur**
  - Une déclaration entre accolades { }, composée de **propriétés** (séparées par des ; )
- ❖ Une **propriété** est constituée de
  - Un **mot-clé**, suivi de ":"
  - La (les) **valeur**(s) associée(s)
- ❖ Exemple : `h1{ font-size:18px; }`

## 2. Les règles d'écriture



- ❖ Les commentaires dans les feuilles de style s'écrivent entre `/*` et `*/`
  - Par exemple : `/* commentaires */`
    - Contrairement aux commentaires HTML de type `<!-- commentaires -->`
- ❖ Commenter son code est toujours conseillé, notamment lorsque les feuilles de style deviennent longues / complexes

## 2. Les règles d'écriture



### Trois façons d'intégration du code css

#### ❖ Données incorporées dans les balises

➤ `<p style="color:red; font-weight:bold;"> Rouge gras</p>`

#### ❖ Données incorporées dans l'entête

➤ `<head> ... <style type="text/css"> h4 { color:green }  
 </style> </head>`

#### ❖ Feuille externe attachée

➤ `méthode recommandée`



## 2. Les règles d'écriture



### Référencement dans un fichier HTML

Exemple:

<head>

<title>Titre de la page</title>

<meta charset="utf-8" />

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

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

</head>

*URL de la ressource liée*

*média auquel s'applique la ressource liée*

*type de contenu auquel le lien fait référence*

*indique la relation qui existe entre le document et la ressource liée*

# 3. Sélecteurs simples



## ❖ Trois types de sélecteurs simples

- les balises
- les classes
- les identificateurs

# 3. Sélecteurs simples



## Les balises

- ❖ Toutes les **balises HTML** peuvent servir de **sélecteurs** à des règles CSS
- ❖ Exemples
  - Pour agir sur les **paragraphes**
    - `p { font-size: 14px; }`
  - Pour agir sur les **liens**
    - `a { font-family: tahoma, arial, verdana; }`

# 3. Sélecteurs simples



## Les classes

### Les classes HTML

- `<p id="p1" class="maclasse">...</p>`
- `<span id="s2" class="maclasse">...</span>`
  - Familles d'éléments librement définies
  - Destinées à l'usage css

# 3. Sélecteurs simples



## Les classes

### ❖ Exemple d'utilisation

➤ `.maclasse { color : blue ; }`

- La règle s'appliquera à **tout élément** HTML de classe "maclasse"

➤ `p.maclasse { color : blue ; }`

- restriction à un élément HTML : La règle s'appliquera à tout **paragraphe** de classe "maclasse"
- (mais pas aux autres éléments de classe "maclasse" : `div, h1, span, table ...`)

# 3. Sélecteurs simples



## Les identificateurs

- ❖ Les **identificateurs** pointent sur un seul élément HTML

- `<h1 id="grandtitre">Mon titre</h1>`

- ❖ Exemple d'utilisation

- `#grandtitre { font-weight: bold ; }`

# 3. Sélecteurs simples



## Quel sélecteur choisir ?

- ❖ Lorsqu'un élément est identifié de manière unique dans un document
  - Utiliser un identificateur (le code est ainsi plus facile à lire et à maintenir)
- ❖ Les classes sont adaptées pour des éléments génériques, appelés à être utilisés **plusieurs fois** dans un même document : « types » de paragraphes, éléments redondants (mise en valeur de portions de texte) ...
  - C'est une forme de factorisation (parmi d'autres)

# 3. Sélecteurs simples



## Sélection par imbrication / hiérarchie

- ❖ Les liens « descendants » d'un élément identifié

- `#monMenu a { color : blue ; }`

- ❖ On peut utiliser plus d'éléments dans la hiérarchie

- `#monMenu li a { color : blue ; }`

- ❖ `*` : sélecteur universel (« joker »)

- `div * a { color : blue ; }`

- tous les liens contenus dans un élément lui-même dans une div



# 4. Sélecteurs avancés



## Sélection par imbrication / hiérarchie

❖ **>** : Descendance directe (1 niveau)

➤ `div > p { color : blue ; }`

- p devient **bleu** seulement si p est fils d'une div

➤ `body > *`

- Tous les éléments descendant directement du corps du document (et uniquement ceux-ci)

# 4. Sélecteurs avancés



## Sélection par succession

❖ **+** indique une succession stricte (dans le fil du code)

➤ `div + p { margin-top : 20px ; }`

■ `p` présente une marge haute de 20 pixels seulement s'il succède **directement** un `div`

● exemple : `<div>...</div> <p>...</p>`

➤ NB : succession != descendance

❖ **~** indique une succession générale (dans le flux)

➤ `div ~ p { margin-top : 20px ; }`

■ `p` présente une marge haute de 20 pixels seulement s'il succède un `div` (**directement ou pas**)

● `<div>...</div> <a>..</a> <p>...</p>`

## 4. Sélecteurs avancés



### Factorisation

#### ❖ Regroupement de sélecteurs

➤ `h1, h2, h3 { /* declaration */ }`

#### ❖ Regroupement de propriétés

➤ `div { border : solid 1px #000000 ; }`

**au lieu de**

➤ `div { border-style : solid ;  
border-width : 1px ;  
border-color : #000000 ; }`

## 4. Sélecteurs avancés



**Factorisation** : regroupement de propriétés

**Background** = raccourci de :

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

**font** = raccourci de :

- font-style
- font-variant
- font-weight
- font-size/line-height
- font-family

**margin** = raccourci de :

- margin-top
- margin-right
- margin-bottom
- margin-left

**Padding** = raccourci de :

- padding-top
- padding-right
- padding-bottom
- padding-left

# 4. Sélecteurs avancés



## Sélection par attribut

### ❖ Sélectionner des éléments par leurs attributs

- `[attribut]`
- `Elt[attribut]`
- `[attribut="valeur"]`

### ❖ Toutes les images ayant l'attribut alt

- `img[alt] {...}`

### ❖ Tous les groupements de 2 cellules

- `td[colspan="2"] {...}`

### ❖ Tous les paragraphes dont le texte est centré

- `p[align=center] {...}`

## 4. Sélecteurs avancés



### Sélection par attribut : opérateurs

- ❖ valeur de l'attribut **commence** par ...
- ❖ valeur de l'attribut **finit** par ...
- ❖ valeur de l'attribut **contient** ...
- ❖ [attribut **^=** "valeur"]
- ❖ [attribut **\$=** "valeur"]
- ❖ [attribut **\*=** "valeur"]

## 4. Sélecteurs avancés



### Pseudo classes

```
selecteur : pseudo-classe { ... }
```

- ❖ Des cas précis, limités
- ❖ Permettent des directives de style sur des éléments spécifiques
- ❖ Exemple :
  - Agir sur la couleur des liens selon leur « état »

# 4. Sélecteurs avancés



## Pseudo classes (exemples)

### ❖ Agir sur les liens

- `a : link` (liens standards)
- `a : visited` (liens déjà visités)
- `a : hover` (liens au survol de souris)
- `a : active` (lien actif)

### ❖ Agir sur le “1er enfant”

- `p : first-child { /*déclaration*/ }`



## 4. Sélecteurs avancés



### Pseudo éléments

```
selecteur :: pseudo-element { ... }
```

- ❖ Agir sur **la première lettre d'un texte** avec la directive de style first-letter

➤ **p** :: first-letter { /\*déclaration\*/ }

- ❖ Agir sur **la première ligne d'un bloc** avec la directive de style first-line

➤ **p** :: first-line { /\*déclaration\*/ }

# 5. TP2



Avant de démarrer votre 2ème TP, voici les principales propriétés CSS

- Propriétés de mise en forme du texte
  - font-family, font-size, font-style, font-variant, text-align, text-shadow...
- Propriétés de couleur et de fond
  - color, background-color, background-image, opacity...
- Propriétés des listes
  - list-style-type, list-style-position, list-style-image...
- Propriétés des tableaux
  - border, border-collapse, vertical-align, caption-side,...

# 5. TP2



- Propriétés des liens
  - (pseudo-classes)
- Propriétés des boîtes
  - width, height, margin, margin-left, margin-right,... padding, border,
- Propriétés de positionnement et d'affichage
  - display, overflow, float, clear, position,...

## 5. TP2



- Liste complète des sélecteurs

[https://www.w3schools.com/cssref/css\\_selectors.asp](https://www.w3schools.com/cssref/css_selectors.asp)

- TP2 sur Chamilo