

INTRODUCTION AU DÉVELOPPEMENT WEB

Vendredi 3 juin

Option Informatique
Ecole Alsacienne

PLAN

1. Un peu de vocabulaire
2. Afficher une page web
3. HTML : La structure et le contenu
4. CSS : la mise en forme
5. Clients et serveurs
6. JS : de nouvelles interactions
7. Mise en pratique

UN PEU DE VOCABULAIRE

LE WEB, LE NET, ETC.

Comment définiriez-vous les termes suivants ?

- Web

- Abréviation courante de World Wide Web (littéralement "toile d'araignée mondiale")
- Inventé principalement par Tim Berners-Lee en 1990
- Une des applications d'Internet, dédiée à la publication et à la consultation des documents



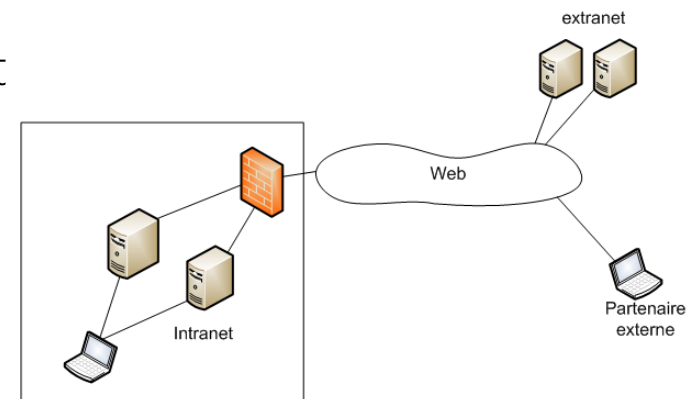
- Net

- Abréviation courante pour Internet
- Descendant du projet ARPANET, lancé par le DARPA en 1969
- Réseau mondial de communication (ou plutôt réseau de réseaux)
- Applications : World Wide Web, messagerie électronique, chats, P2P, etc.



LE WEB, LE NET, ETC.

- Extranet
 - Réseau de télécommunication dédié à une organisation et accessible depuis l'extérieur de celle-ci
 - Exemples d'utilisation : travail à domicile, partage de documents, etc.
- Intranet
 - Réseau de télécommunication dédié à une entreprise et accessible uniquement depuis celle-ci
 - Avantages : sécurité des données, rapidité des échanges
- .NET (prononcé "dot net")
 - Ensemble d'outils développés par Microsoft pour les applications destinées à Internet
 - Exemples : C#, ASP.NET, etc.



LES ACRONYMES DU WEB

- HTML

- Hypertext Markup Language
- Invité au début des années 90, et très lié au World Wide Web
- Objectif : écrire des documents "hypertextuels", reliés les uns aux autres par des "hyperliens"
- HTML5 : Nouvelle spécification initiée en 2007 et finalisé en 2014



- CSS

- Cascading Style Sheets (en français : « feuilles de style en cascade »)
- Introduit au milieu des années 90
- Largement adopté et supporté dans les années 2000
- Objectif : mettre en forme les documents "hypertextuels"
- CSS3 : Nouvelle spécification initiée en 1999, et toujours en cours



LES ACRONYMES DU WEB



- JS

- JavaScript
- Langage de script principalement employé pour des pages web
- Quelle est la difference entre Java et JavaScript ?
Java and Javascript are similar like Car and Carpet are similar.
- Créé en 1995 par Brendan Eich (en 10 jours)
- Basé sur la spécification ECMAScript (sur laquelle sont également basés le JScript de Microsoft et l'ActionScript d'Adobe)

- PHP

- PHP: Hypertext Preprocessor (acronyme récursif)
- Créé par Rasmus Lerdorf
- Principalement utilisé pour produire des pages web "dynamiques"
- Dernière version en date : 7.0 (sortie le 3 décembre 2015)

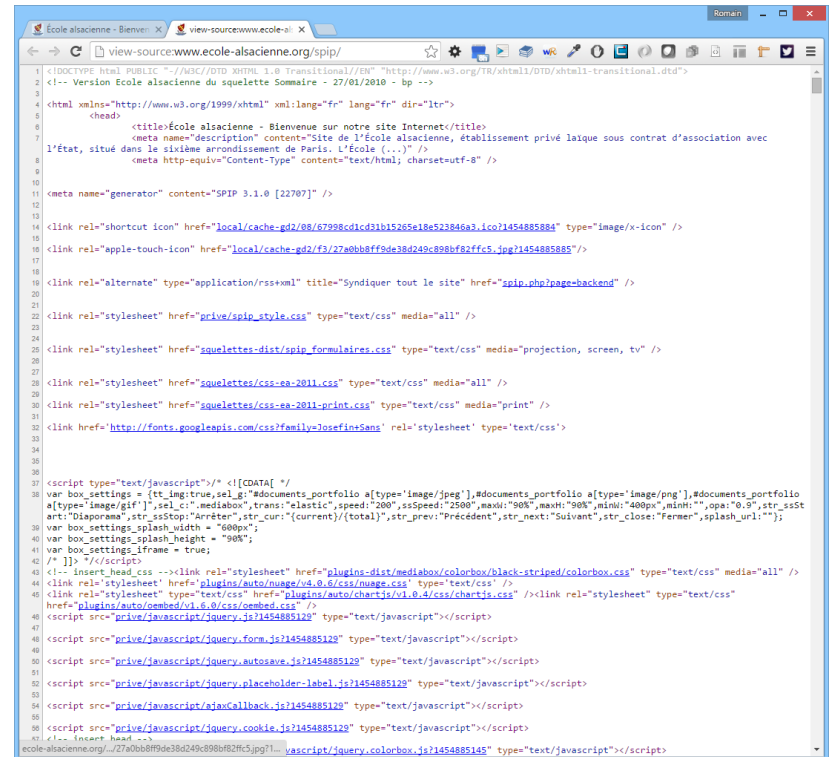


LES ACRONYMES DU WEB

- Il existe beaucoup d'autres acronymes sur Internet:
 - DOM : Document Object Model
 - XML : Extensible Markup Language
 - SQL : Structured Query Language
 - JSON : JavaScript Object Notation
 - MIME : Multipurpose Internet Mail Extensions
 - HTTP : Hypertext Transfer Protocol
 - TCP/IP : Transfer Control Protocol / Internet Protocol
 - FTP : File Transfert Protocol
 - SMTP : Simple Mail Transfer Protocol
 - SSH : Secure Shell
 - etc.

AFFICHER UNE PAGE WEB

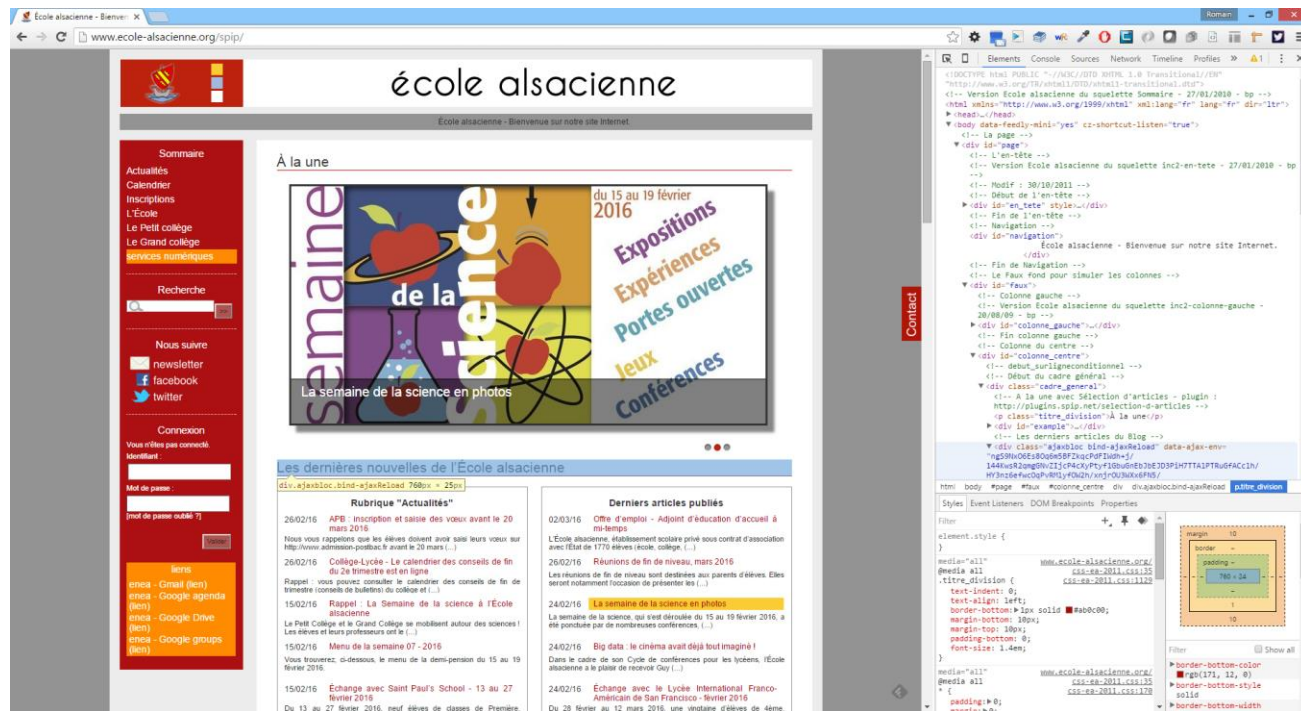
OBSERVONS UNE PAGE WEB



- Il est possible d'afficher le code source d'une page web dans un navigateur (en général avec le raccourci Ctrl+U)

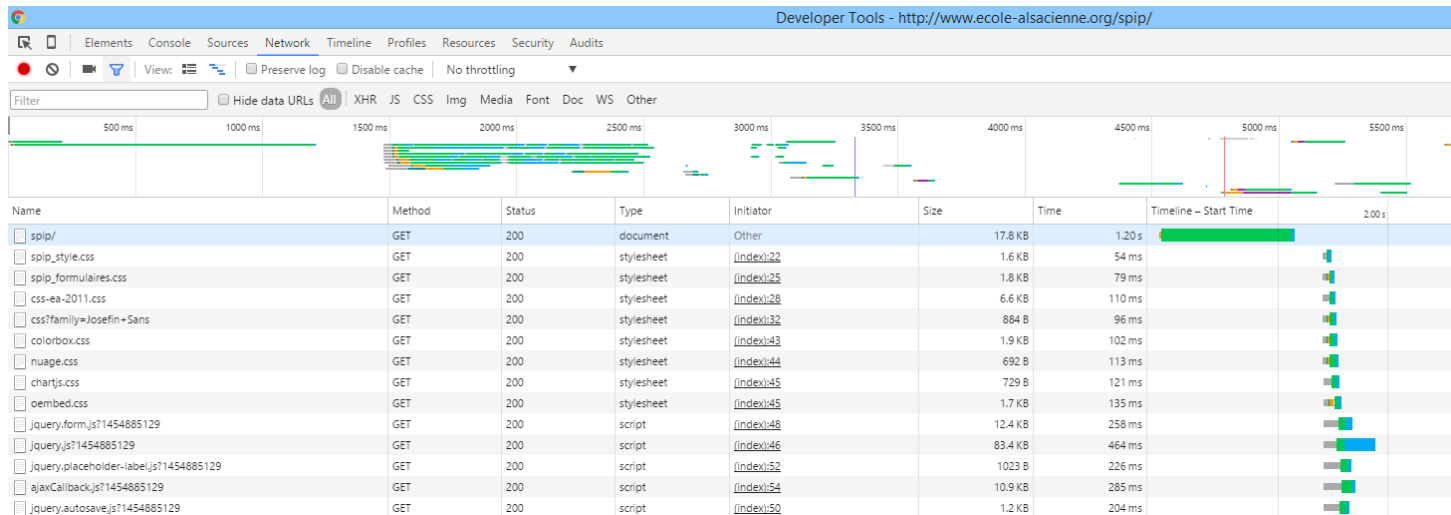
MODE DÉVELOPPEUR

- Il n'est pas toujours évident de s'y retrouver lorsqu'on regarde le code source d'une page
 - Plusieurs navigateurs proposent désormais un mode « développeur »
 - Il suffit en général d'appuyer sur la touche F12



DE NOMBREUX FICHIERS

- Ce mode développeur permet notamment d'observer les « requêtes » effectuées par le navigateur.
 - Il s'agit généralement de la vue "Réseau" ou "Network"
- On voit ainsi que pour afficher une page, le navigateur doit parfois récupérer de nombreux fichiers :

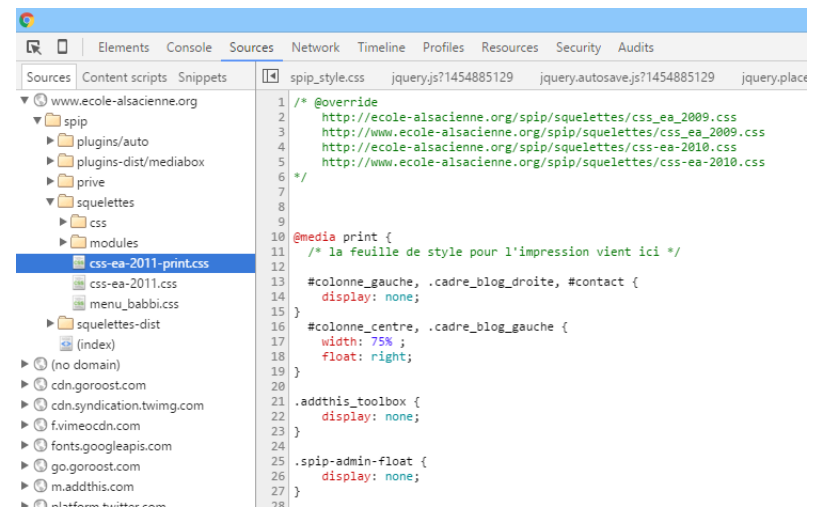
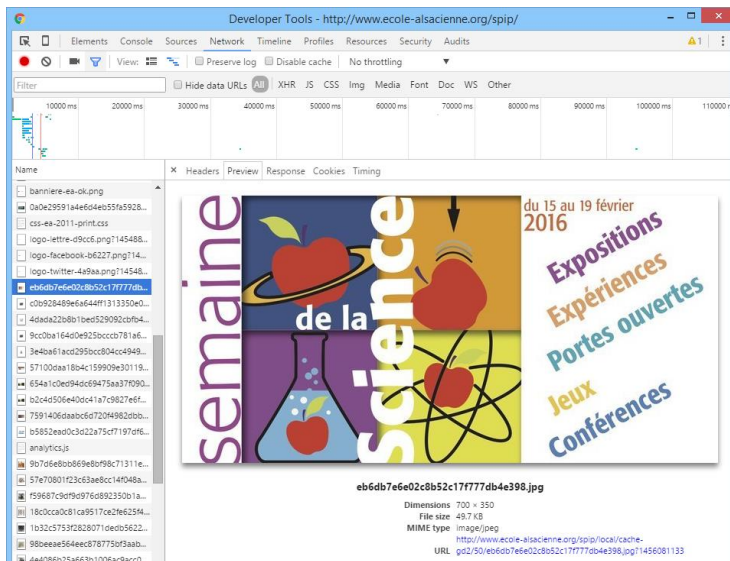


DE NOMBREUX FICHIERS

- Lorsqu'un navigateur accède à une page internet, il télécharge en fait de nombreux fichiers :
 - Un ou plusieurs fichiers correspondant au contenu de la page (HTML, PHP)
 - Des feuilles de styles (fichiers CSS)
 - Des scripts (fichiers JS)
 - Des images
 - etc.
- Le navigateur a donc un rôle d'interpréteur :
 1. Il récupère les différents éléments qui composent la page
 2. Il utilise tous ces éléments pour afficher la page (de manière graphique) à l'utilisateur

DE NOMBREUX FICHIERS

- Il est généralement possible de visualiser ces fichiers sous-jacents qui sont utilisés pour afficher la page :
 - On peut notamment le faire depuis les vues "Network" et "Sources"
 - Cela peut s'avérer très pratique en cas de problème



NAVIGATEURS INTERNET

- Il existe une grande variété de navigateurs internet
 - Plusieurs poids lourds
 - Navigateurs mobiles
- Plusieurs versions d'un même navigateur peuvent coexister

- Répartition des usages :

<http://gs.statcounter.com/>



NAVIGATEURS INTERNET

- Il peut exister des différences importantes entre les navigateurs, et entre les versions d'un même navigateur
 - En particulier, certaines fonctionnalités ne sont disponibles que sur les navigateurs dits « récents »
 - Cela peut avoir des conséquences non négligeables dans la pratique (temps de développement et de maintenance plus importants)



NAVIGATEURS INTERNET

Comment savoir si une fonctionnalité est supportée par un navigateur en particulier ?

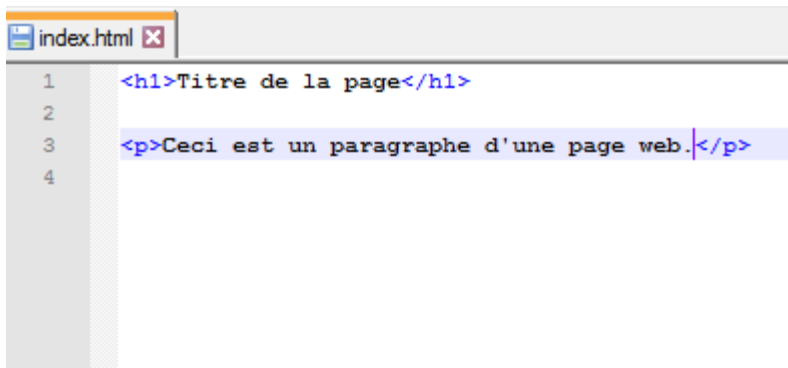


<http://caniuse.com/>

HTML : LA STRUCTURE ET LE CONTENU

LA BRIQUE CENTRALE : STRUCTURE ET CONTENU

- Le code HTML est la brique centrale d'une page internet
 - Un fichier `.html` suffit pour faire une page web



```
index.html x
1 <h1>Titre de la page</h1>
2
3 <p>Ceci est un paragraphe d'une page web.</p>
4
```



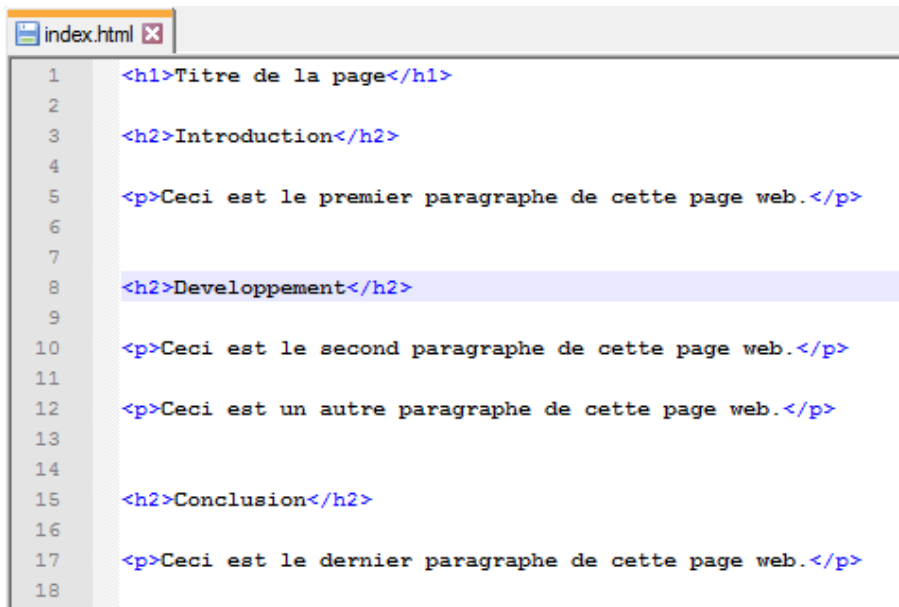
- Le code HTML contient
 - La structure de la page internet (titres, paragraphes, listes, etc.)
 - Le contenu lui-même

BALISES HTML



- Le langage HTML repose sur la notion de **balise**
- Ces balises sont identifiées par des chevrons
 - Soit sous la forme d'un couple de balises `<nom> ... </nom>`
 - Soit sous la forme d'une balise unique `<nom ... />`
- Une balise représente un composant d'une page internet
 - Un titre `<h1>`
 - Un paragraphe `<p>`
 - Une image `<img>`
 - Une liste à puces `<ul>`
 - etc.

BALISES HTML : PREMIER EXEMPLE



A screenshot of a code editor window titled 'index.html'. The editor contains 18 lines of HTML code. Lines 1-3 define the main title and a section header. Lines 4-6 contain a paragraph. Lines 7-9 define another section header. Lines 10-12 contain two paragraphs. Lines 13-15 define a third section header. Lines 16-18 contain a final paragraph. The line numbers 1 through 18 are listed on the left side of the editor.

```
1 <h1>Titre de la page</h1>
2
3 <h2>Introduction</h2>
4
5 <p>Ceci est le premier paragraphe de cette page web.</p>
6
7
8 <h2>Developpement</h2>
9
10 <p>Ceci est le second paragraphe de cette page web.</p>
11
12 <p>Ceci est un autre paragraphe de cette page web.</p>
13
14
15 <h2>Conclusion</h2>
16
17 <p>Ceci est le dernier paragraphe de cette page web.</p>
18
```

Titre de la page

Introduction

Ceci est le premier paragraphe de cette page web.

Developpement

Ceci est le second paragraphe de cette page web.

Ceci est un autre paragraphe de cette page web.

Conclusion

Ceci est le dernier paragraphe de cette page web.

DES BALISES DANS DES BALISES...



- Une balise peut elle-même contenir d'autres balises
- Par exemple, une balise `<ul>` représente une liste à puces, et chaque élément de cette liste est identifié par une balise `<li>`

```
index.html x
1  <h1>Titre de la page</h1>
2
3  <h2>Introduction</h2>
4
5  <p>Ceci est le premier paragraphe de cette page web.</p>
6
7  <ul>
8    <li>Point A</li>
9    <li>Point B</li>
10   <li>Point C</li>
11 </ul>
12
13
14 <h2>Developpement</h2>
15
16 <p>Ceci est le second paragraphe de cette page web.</p>
17
18
19
20 <h2>Conclusion</h2>
21
22 <p>Ceci est le dernier paragraphe de cette page web.</p>
23
```

Titre de la page

Introduction

Ceci est le premier paragraphe de cette page web.

- Point A
- Point B
- Point C

Developpement

Ceci est le second paragraphe de cette page web.

Conclusion

Ceci est le dernier paragraphe de cette page web.

DES BALISES DANS DES BALISES...



- La balise `<div>` est un conteneur générique, généralement utilisé pour définir plusieurs « zones » dans la page

```
index.html
1 <div id="contenu">
2   <h1>Titre de la page</h1>
3
4   <div id="section1">
5     <h2>Introduction</h2>
6     <p>Ceci est le premier paragraphe de cette page web.</p>
7     <ul>
8       <li>Point A</li>
9       <li>Point B</li>
10      <li>Point C</li>
11    </ul>
12  </div>
13
14  <div id="section2">
15    <h2>Developpement</h2>
16    <p>Ceci est le second paragraphe de cette page web.</p>
17  </div>
18
19  <div id="section3">
20    <h2>Conclusion</h2>
21    <p>Ceci est le dernier paragraphe de cette page web.</p>
22  </div>
23 </div>
```

Titre de la page

Introduction

Ceci est le premier paragraphe de cette page web.

- Point A
- Point B
- Point C

Developpement

Ceci est le second paragraphe de cette page web.

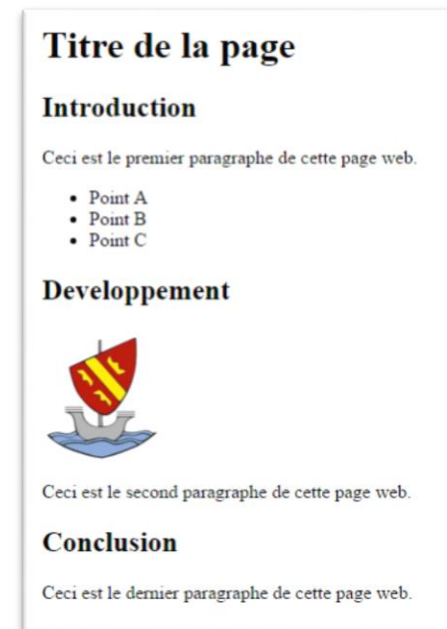
Conclusion

Ceci est le dernier paragraphe de cette page web.

ET LES IMAGES ?

- La balise `<img>` permet d'afficher des images sur une page internet.
- La source est indiquée dans l'attribut `src` : c'est le chemin à suivre pour obtenir le fichier source.

```
index.html x
1 <div id="contenu">
2   <h1>Titre de la page</h1>
3
4   <div id="section1">
5     <h2>Introduction</h2>
6     <p>Ceci est le premier paragraphe de cette page web.</p>
7     <ul>
8       <li>Point A</li>
9       <li>Point B</li>
10      <li>Point C</li>
11    </ul>
12  </div>
13
14  <div id="section2">
15    <h2>Developpement</h2>
16    
17    <p>Ceci est le second paragraphe de cette page web.</p>
18  </div>
19
20  <div id="section3">
21    <h2>Conclusion</h2>
22    <p>Ceci est le dernier paragraphe de cette page web.</p>
23  </div>
24 </div>
```



CSS : LA MISE EN FORME

A QUOI SERT UN FICHIER CSS ?



Les fichiers CSS permettent de mettre en forme le contenu déclaré dans les fichiers HTML.

■■■■■ CSS : la mise en forme

PREMIER EXEMPLE CSS

```
index.html x layout.css x
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link rel="stylesheet" href="./layout.css"/>
5   </head>
6
7   <body>
8     <h1>Titre de la page</h1>
9
10    <p>Le CSS permet de modifier la mise en forme du
11    contenu HTML.</p>
12  </body>
</html>
```

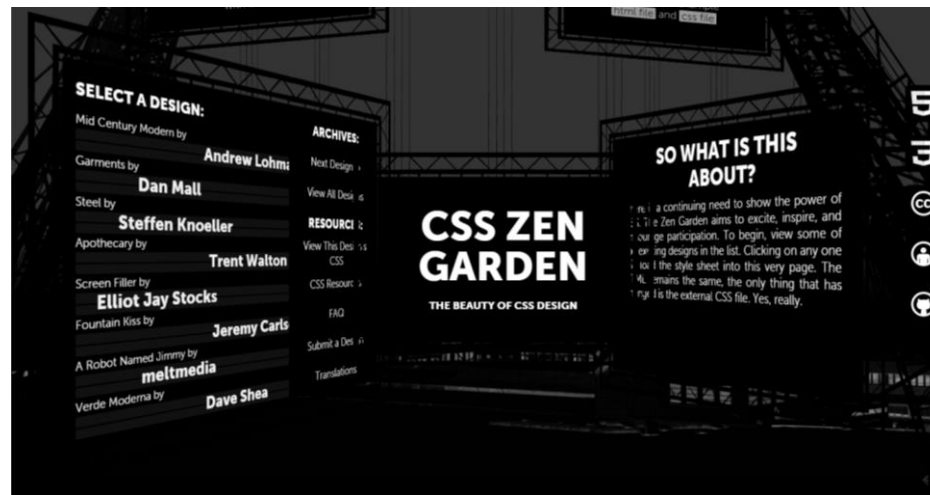
```
index.html x layout.css x
1 h1 {
2   color: red;
3   font-style: italic;
4   font-family: Arial;
5 }
6
7 p {
8   background: lightblue;
9   border: 5px blue solid;
10  padding: 10px;
11 }
12
```

Titre de la page

Le CSS permet de modifier la mise en forme du contenu HTML.

QUE PEUT-ON FAIRE EN CSS ?

- Il est possible de faire énormément de choses grâce au CSS
 - Mise en forme du texte (police, taille, couleur, ...)
 - Gestion des blocs (bordures, couleurs de fond, ...)
 - Taille des blocs (largeur, hauteur, positionnement, ...)
 - etc.



- Exemples : <http://www.csszengarden.com/>

CASCADING ?

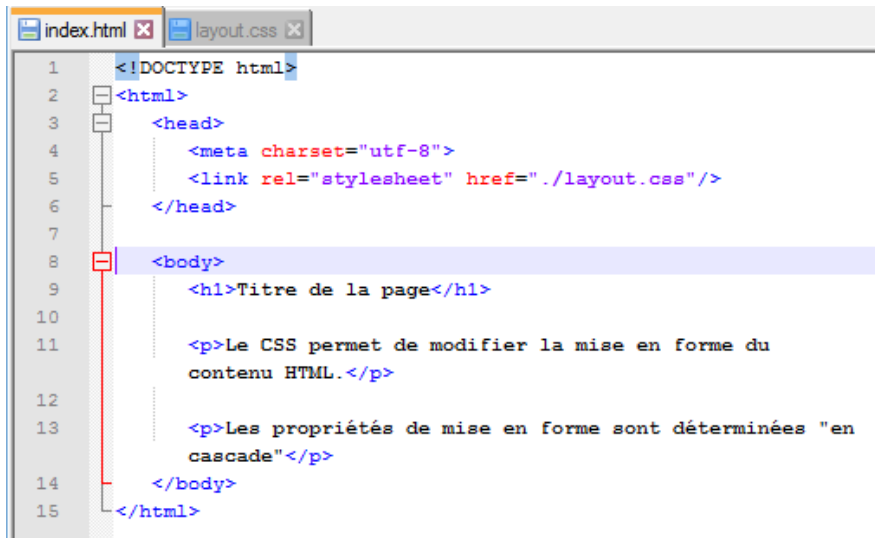
Selon vous, que signifie le C dans l'acronyme CSS ?



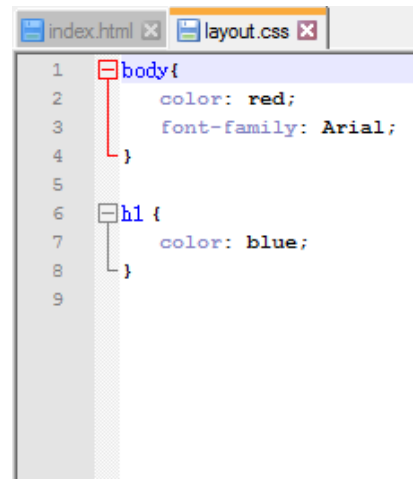
Cascading

CASCADING ?

- On parle de feuilles de styles en cascade (cascading stylesheets en anglais) car les règles de mise en forme sont appliquées « en cascade »
 - Si une propriété X est définie pour un élément Y, cette définition est aussi valable pour tous les éléments inclus dans cet élément Y.
 - Il est cependant possible de redéfinir la propriété X pour un certain élément Z inclus dans l'élément Y



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="./layout.css"/>
6   </head>
7
8   <body>
9     <h1>Titre de la page</h1>
10
11     <p>Le CSS permet de modifier la mise en forme du
12     contenu HTML.</p>
13
14     <p>Les propriétés de mise en forme sont déterminées "en
15     cascade"</p>
16   </body>
17 </html>
```



```
1 body{
2   color: red;
3   font-family: Arial;
4 }
5
6 h1 {
7   color: blue;
8 }
9
```

Titre de la page

Le CSS permet de modifier la mise en forme du contenu HTML.

Les propriétés de mise en forme sont déterminées "en cascade"

LIENS AVEC LE CODE HTML

- Il est ainsi possible de définir la mise en forme par défaut des balises HTML
 - `<h1>`
 - `<p>`
 - etc.
- Mais le CSS ne se limite pas à ces balises « standard » !
- La puissance du CSS repose notamment sur les deux notions suivantes :
 - les identifiants
 - les classes

LES IDENTIFIANTS

- Question : *Comment écrire tous les paragraphes en bleu, sauf le premier en rouge ?*



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="./layout.css"/>
6   </head>
7
8   <body>
9     <h1>Titre de la page</h1>
10
11    <p id="paragraphe1">Le CSS permet de modifier la mise
12      en forme du contenu HTML.</p>
13
14    <p>Les propriétés de mise en forme sont déterminées "en
15      cascade".</p>
16
17    <p>Les identifiants permettent de cibler un élément en
18      particulier</p>
19  </body>
20 </html>
```

```
1 body {
2   font-family: "Arial";
3 }
4
5 p {
6   color: blue;
7 }
8
9 #paragraphe1 {
10   color: red;
11 }
12
```

- Pour toute balise HTML, on peut définir l'attribut **id** pour l'identifier de manière unique
 - Cet identifiant peut être utilisé dans un fichier CSS
 - On utilise alors le symbole # suivi de l'identifiant lui-même

LES CLASSES

- Question : *Comment écrire en bleu les paragraphes pairs, et en rouge les paragraphes impairs ?*

Titre de la page

Le CSS permet de modifier la mise en forme du contenu HTML.

Les propriétés de mise en forme sont déterminées "en cascade".

Les identifiants permettent de cibler un élément en particulier.

Les classes permettent d'utiliser la même mise en forme pour plusieurs éléments.

```
1 body {
2     font-family: "Arial";
3 }
4
5 .pair {
6     color: blue;
7 }
8
9 .impair {
10     color: red;
11 }
12
```

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="../layout.css"/>
6 </head>
7
8 <body>
9     <h1>Titre de la page</h1>
10
11     <p class="pair">Le CSS permet de modifier la mise en
12     forme du contenu HTML.</p>
13
14     <p class="impair">Les propriétés de mise en forme sont
15     déterminées "en cascade".</p>
16
17     <p class="pair">Les identifiants permettent de cibler
18     un élément en particulier.</p>
19
20     <p class="impair">Les classes permettent d'utiliser la
21     même mise en forme pour plusieurs éléments.</p>
22 </body>
23 </html>
```

- Pour toute balise HTML, on peut définir l'attribut **class**, qui indique donc à quelle « classe » d'élément il s'agit.
 - Les classes sont avant tout utilisées dans les fichiers CSS
 - On utilise le symbole `.` suivi de l'identifiant lui-même
 - Contrairement à l'id, plusieurs éléments peuvent avoir la même classe.
 - Un élément peut appartenir à plusieurs classes

LA PUISSANCE DU CSS

- Les fichiers CSS offrent un large éventail de possibilité pour sélectionner les éléments à mettre en forme
 - Comportement au survol

```
#bouton:hover {  
    color: red;  
}
```
 - Sélection multiple

```
.boutonRond, .boutonCarre {  
    font-weight: bold;  
}
```
 - Sélection sur plusieurs niveaux

```
#conteneur p {  
    font-size: 20px;  
}
```
- Il existe également des outils comme LESS ou SASS qui enrichissent encore ces possibilités (par exemple, définir une couleur comme variable)

CLIENTS ET SERVEURS

EXEMPLE : COMMENT S'IDENTIFIER SUR UNE PAGE INTERNET ?

The screenshot shows the homepage of 'école alsacienne'. On the left is a sidebar with a red background containing a navigation menu, a search bar, social media links, and a login section. The main content area on the right has a white background and features a large image of a wall covered in small photos, a section for the latest news, and a grid of recent articles.

Sidebar (Left):

- Logo: A stylized sailboat on a red background.
- Menu: Sommaire, Actualités, Calendrier, Inscriptions, L'Ecole, Le Petit collège, Le Grand collège, services numériques (highlighted).
- Recherche: A search bar with a 'Rechercher' button.
- Nous suivre: newsletter, facebook, twitter.
- Connexion: Vous n'êtes pas connecté. Identifiant: [input], Mot de passe: [input], [mot de passe oublié ?], Valider.
- Liens: home, email - Gmail (new), agenda - Google agenda (new).

Main Content Area (Right):

- Header: école alsacienne, Ecole alsacienne - Bienvenue sur notre site internet.
- À la une: A large image of a wall covered in small photos. Text overlay: Big data : le cinéma avait déjà tout imaginé ! Conférence - lundi 14 mars 2016 à 17h.
- Les dernières nouvelles de l'École alsacienne: A section with a red header containing a list of recent news items.

Recent News Items:

Rubrique "Actualités"	Derniers articles publiés
28/02/16: AFB - inscription et saisie des vœux avant le 28 mars 2016 Nous vous remercions pour les vœux de 2016-2017. Nous vous invitons à nous adresser vos vœux avant le 28 mars.	02/03/16: Choix d'emploi - Adjoint d'éducation d'accueil à mi-temps L'École Alsacienne, établissement scolaire privé sous contrat d'association avec l'Etat de 1770 élèves-école-collège.
28/02/16: Collège-Lycée - Le calendrier des conseils de fin de trimestre est en ligne Rappel : vous pouvez consulter le calendrier des conseils de fin de trimestre (conseils de parents, du collège et du lycée).	28/02/16: Réunions de fin de niveau, mars 2016 Les réunions de fin de niveau sont destinées aux parents d'élèves. Elles seront également l'occasion de présenter les...
15/02/16: Rappel - La Semaine de la science à l'École	24/02/16: La semaine de la science en photos

Selon vous, que se passe-t-il lors qu'un utilisateur essaye de se connecter à un site web ?

CLIENT ET SERVEUR

- En informatique, un serveur est un dispositif qui offre des services en répondant automatiquement à des requêtes.



- Les serveurs sont au cœur de la quasi-totalité des dispositifs numériques aujourd'hui !
 - On estime à plusieurs millions le nombre de serveurs de Google



A QUOI SERT UN SERVEUR ?

- Utiliser un serveur a de nombreux avantages :
 - Les informations et traitements sensibles ne sont pas accessibles à l'utilisateur : ils se déroulent sur le serveur
 - Plus généralement, il est alors possible de n'envoyer à l'utilisateur que les informations dont il a besoin
 - Plusieurs utilisateurs peuvent accéder simultanément au même serveur, aux mêmes services
 - etc.
- Un serveur apporte aussi quelques contraintes
 - Ce serveur doit être accessible pour que l'application fonctionne
 - Un temps de communication est nécessaire à chaque échange entre le client et le serveur

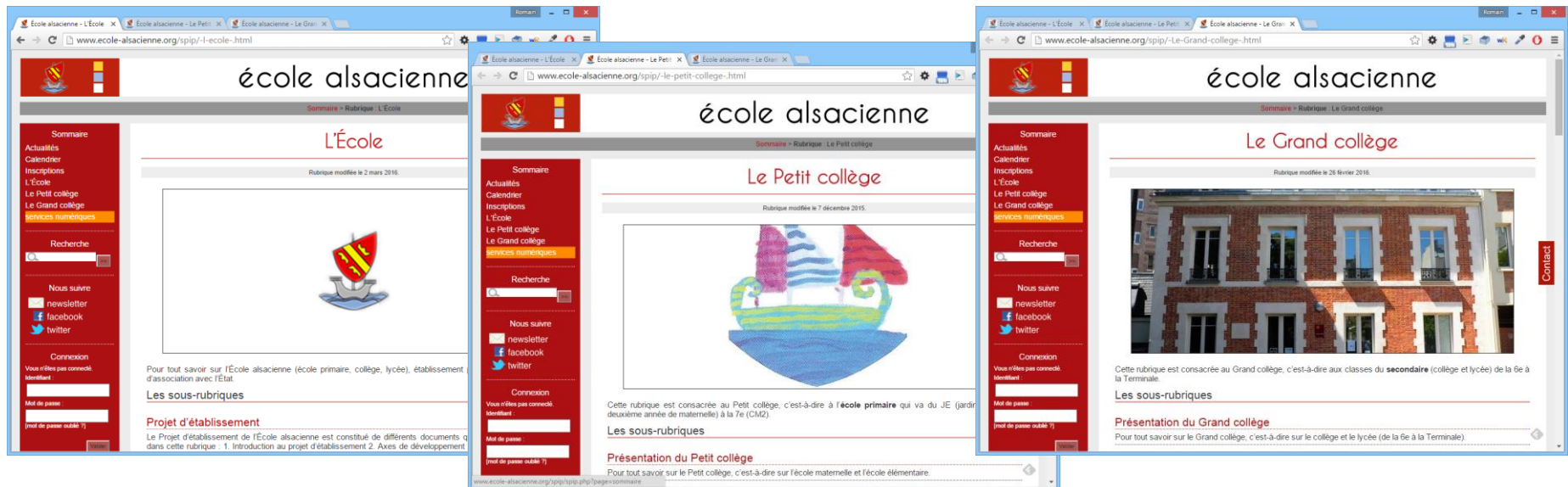
UTILISATEURS MULTIPLES

- L'un des principaux avantages d'un serveur est qu'il permet de mettre en contact de nombreux utilisateurs



CONSTRUIRE UNE PAGE PAR MORCEAUX

- Un serveur permet également de construire une page internet « morceau par morceau ».
- Par exemple, on pourra écrire un entête une seule fois et l'inclure à la volée au moment de restituer les pages :



UNE APPLICATION DÉDIÉE

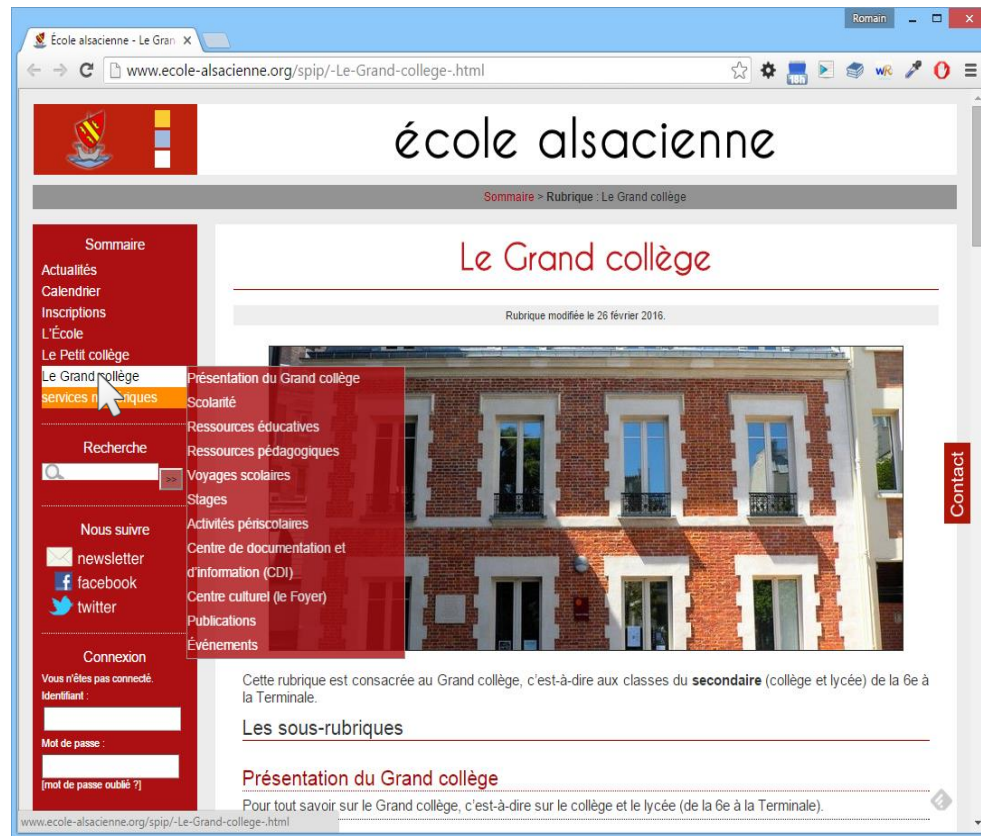
- Un serveur requiert donc une application dédiée, qui s'exécute généralement sur une machine distante.



- Il existe différents langages pour réaliser la partie serveur d'un projet Web : PHP, ASP.NET, etc.
- Remarque : il est possible que le serveur s'exécute sur la même machine qu'un utilisateur (mais si l'utilisateur éteint sa machine, le serveur est coupé pour tout le monde)

JS : DE NOUVELLES INTERACTIONS

A QUOI SERT UN FICHER JS ?



Selon vous, comment peut-on afficher ou masquer des éléments sur une page internet suite à un clic ou au survol d'un élément ?

■■■■■ JS : de nouvelles interactions

LE LANGAGE JAVASCRIPT

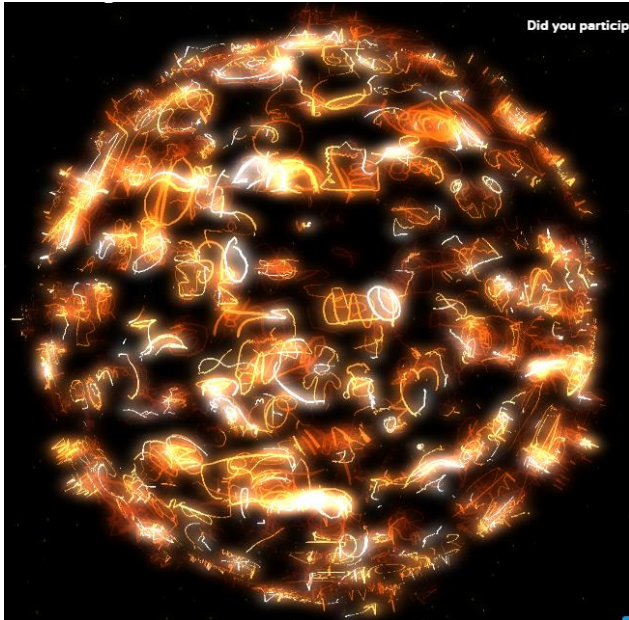


- JavaScript est un langage de scripts, multiplateforme et orienté objet, créé par Brendan Eich en 1995
- *Java and Javascript are similar like car and carpet are similar*
- JavaScript est principalement utilisé côté client, où il permet notamment de
 - déplacer / afficher / masquer des éléments sur la page
 - créer de nouveaux éléments et éditer les éléments existants
 - réagir aux actions de l'utilisateur (clic, scroll, etc.)
- JavaScript peut également être utilisé côté serveur

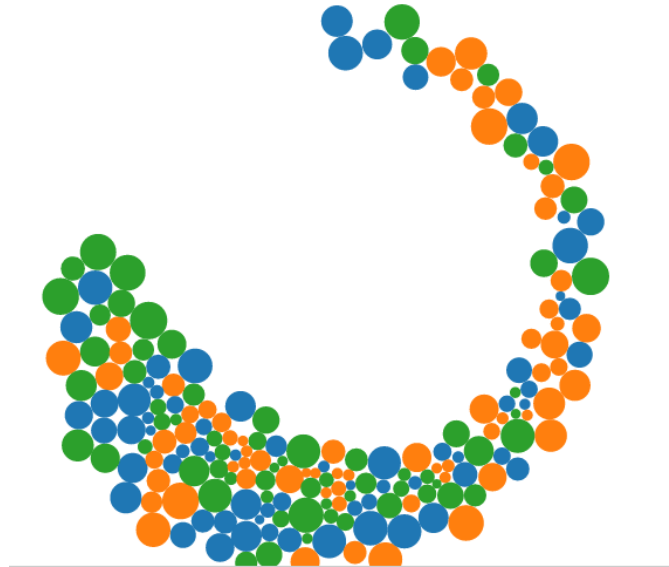
DE NOMBREUSES BIBLIOTHÈQUES

- La communauté JavaScript est très active, et de nombreuses bibliothèques ont été mises à la disposition du grand public
- Quelques exemples :
 - jQuery : manipuler simplement les éléments du HTML
 - Underscore : rechercher efficacement dans des tableaux
 - Angular.js : créer des interactions fortes côté client
 - Moment.js : gérer des dates et des durées
 - d3 : mettre à jour le code HTML selon des jeux de données
 - Three.js : créer des applications en 3D

EXEMPLES



<http://threejs.org/>



<https://d3js.org/>

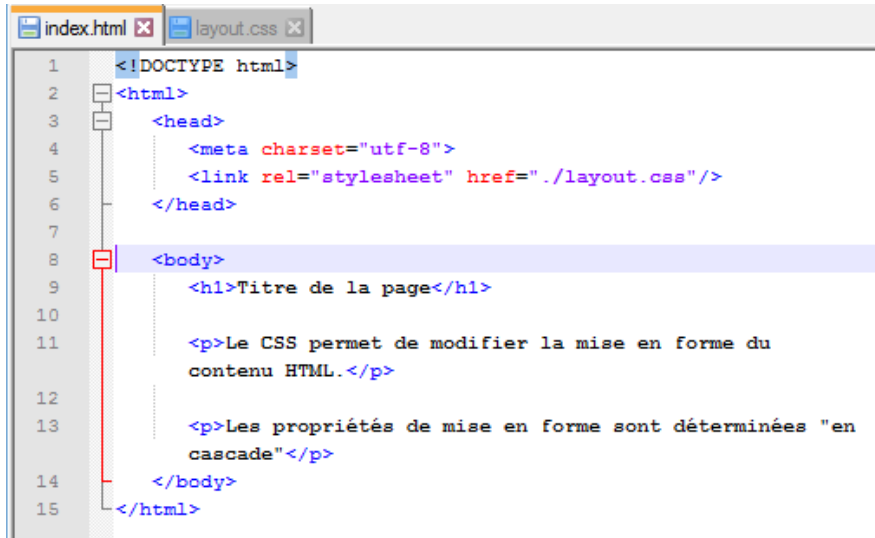


<https://angularjs.org/>

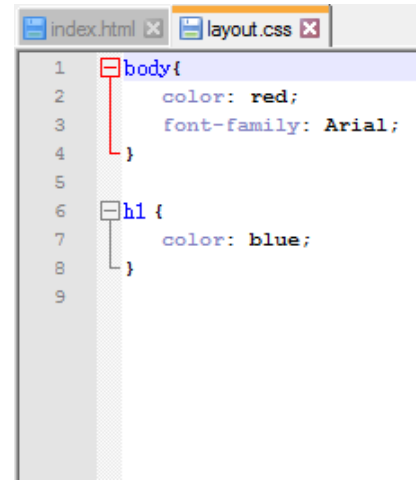
..... JS : de nouvelles interactions

MISE EN PRATIQUE

QUE FAUT-IL POUR RÉALISER UN SITE INTERNET ?



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="./layout.css"/>
6   </head>
7   <body>
8     <h1>Titre de la page</h1>
9
10    <p>Le CSS permet de modifier la mise en forme du
11    contenu HTML.</p>
12
13    <p>Les propriétés de mise en forme sont déterminées "en
14    cascade"</p>
15  </body>
16 </html>
```



```
1 body{
2   color: red;
3   font-family: Arial;
4 }
5
6 h1 {
7   color: blue;
8 }
9
```

Titre de la page

Le CSS permet de modifier la mise en forme du contenu HTML.

Les propriétés de mise en forme sont déterminées "en cascade"

- Un simple bloc-notes suffit !
- Mais un bon environnement de travail permet de travailler de façon plus efficace et plus agréable :
 - Editeur de texte : Notepad++, Visual Studio, SublimeText, etc.
 - Serveur local : WampServer, EasyPHP, IIS, etc.

..... Mise en pratique

ET POUR LE SERVEUR ?

Un serveur requiert une application dédiée, qui s'exécute généralement sur une machine distante.

- La plupart des fournisseurs d'accès internet offre désormais la possibilité de créer des "pages perso"
- De nombreux hébergeurs proposent également l'hébergement de pages personnelles
- Remarque : ces offres "gratuites" sont généralement suffisantes pour de petits projets, mais elles n'offrent pas les mêmes garanties que les offres payantes.

UN VASTE ÉVENTAIL DE COMPÉTENCES

- La réalisation d'un site internet peut nécessiter un vaste éventail de compétences :
 - Graphiste (maquettes graphiques)
 - Rédacteur (contenus textuels)
 - Développeur "front" (code HTML et interactions côté client)
 - Développeur "back" (traitements côté serveur et base de données)
 - Ergonome
 - etc.
- Cependant, il est rare qu'une seule personne soit experte dans tous ces domaines !



UN VASTE ÉVENTAIL DE COMPÉTENCES

- Pour démocratiser le développement Web, plusieurs outils "clef en main" ont vu le jour, afin de permettre à un utilisateur même néophyte de réaliser un site internet
- En fonction du projet et du degré de personnalisation voulu, on privilégiera l'un ou l'autre de ces outils :
 - Moteur de blog : Tumblr, Blogger, etc.
 - Système de gestion de contenus (CMS) : MediaWiki, WordPress, phpBB, Drupal, etc.
 - Panel de composants graphiques : Bootstrap, Pure, etc.
 - Thèmes graphiques pour les outils précédents
 - etc.

PROCHAINE SÉANCE

Vendredi 10 juin

DERNIÈRE SÉANCE

