



Ressources pour le cycle terminal général et technologique

Informatique et Sciences du Numérique

Créer des pages « Web »

Ces documents peuvent être utilisés et modifiés librement dans le cadre des activités d'enseignement scolaire, hors exploitation commerciale.

Toute reproduction totale ou partielle à d'autres fins est soumise à une autorisation préalable du Directeur général de l'enseignement scolaire.

La violation de ces dispositions est passible des sanctions édictées à l'article L.335-2 du Code de la propriété intellectuelle.

Juin 2012

Présentation / Créer des pages « Web »

1 / Thème abordé

1.1 Problématique, situation d'accroche

Les sites Internet proposent de nombreux contenus affichés sous forme de pages « Web » reliées entre elles. La complexité de l'organisation de la page et la richesse des contenus (textes et multimédia) doit permettre malgré tout l'affichage sur un nombre varié de supports (écran d'ordinateur ou de smartphone) et ceci au plus grand nombre d'utilisateurs grâce à des systèmes assistés comme le synthétiseur vocal ou le terminal Braille.

Les activités tourneront autour du questionnement suivant:

- Comment décrire une page (contenu et mise en forme) ?
- Comment afficher une page sur des média différents (écrans d'ordinateur ou de terminal mobile) ?
- Comment favoriser l'accessibilité des pages à certaines catégories d'utilisateurs ?
- Comment réaliser la communication entre ces pages puis organiser la circulation des pages sur le Web ?

L'accroche proposée consiste d'abord à **observer** des pages.

Visualisation d'une page HTML d'après la page de démonstration du site www.csszengarden.com

L'affichage est quasiment limité au texte brut ; en regardant attentivement, on voit qu'on a quand même des changements de taille de caractères, du gras, une mise en forme des liens web et un espacement vertical entre paragraphes).

Le même site, après mise en forme grâce à une feuille de style. Avec l'extension Firebug du navigateur Firefox, on visualise directement l'effet d'une modification d'un paramètre du fichier CSS sur la mise en forme de la page.

```
#pageHeader h1 {
background: url("/001/h1-fr.gif") no-repeat scroll left top transparent;
float: right;
height: 87px;
margin-top: 10px;
width: 219px;
}
```

1.2 Frontières de l'étude et prolongements possibles

L'objectif n'est pas l'apprentissage complet des langages HTML/CSS. Il faudra veiller à ce que cette partie reste suffisamment modeste avec un codage réalisable « à la main »¹.

Le dynamisme des pages et le fait que le langage HTML sert de cible à la génération de pages par programmation (langage serveur) peut être évoqué dans son principe mais ne fait pas l'objet d'activités dans cette séquence ; les activités menées concernent donc uniquement les pages Web dites « statiques ».

Quelques prolongements se présentent naturellement :

- gestion de l'interactivité avec l'utilisateur (principe des pages « dynamiques », langages serveur)

puis autour de la sécurité sur Internet :

- authentification des sites Web: qui est réellement le propriétaire d'une page Web ? (voir la technique du « hameçonnage »)
- sécurité des échanges sur le Web (systèmes SSL et TLS, protocole HTTPS).

2 / Objectifs pédagogiques

2.1 Prérequis

Avant d'aborder le codage en langage HTML, il est préférable que l'élève possède déjà une expérience de la programmation (compétences en écriture et mise au point d'un programme, respect de la syntaxe etc.). Les activités de création de pages Web peuvent être donc menées en milieu d'année.

2.2 Éléments du programme

Contenus

Langage de description HTML.

Compétences et capacités

Décrire et expliquer une situation, un système ou un programme :

- Classer des informations notamment sous forme d'une arborescence ;
- Décrire une situation d'adressage sur un type de réseau particulier ;
- Analyser des pages Web (statiques) simples.

Concevoir et réaliser une solution informatique en réponse à un problème :

- Mettre un programme au point en le testant, en l'instrumentant ;
- Créer et analyser une page Web en langage HTML ;
- Créer (« à la main ») des pages Web simples et les déposer sur un serveur local.

Communiquer à l'écrit et à l'oral :

- Comprendre l'intérêt de la séparation du contenu et de la mise en forme grâce à des pages de style ;
- Évaluer la qualité des pages du point de vue de la correction syntaxique et de l'efficacité du message.

Faire un usage responsable des sciences du numérique :

- Mettre en ligne un site Web sur un serveur distant identifié et identifier les problèmes de sécurité.

3 / Modalités de mise en œuvre

3.1 Durée prévue pour la partie se déroulant en classe

Trois-quatre séances de 2h puis un mini-projet.

3.2 Type de l'animation

Classe entière avec ordinateur et vidéoprojecteur pour introduire la séquence.

Activités en TP en individuel ou par deux.

3.3 Projet

Les activités menées devront permettre la réalisation en autonomie de projets individuels autour de la réalisation de pages Web simples.

¹ Attention à l'utilisation d'éditeurs spécialisés dits « wysiwyg » ou « d'édition visuelle » permettant l'export automatique en HTML mais au code souvent lourd, non-standard et complexe.

3.4 Recherches documentaires

La recherche documentaire concerne les aspects sociétaux et juridiques sur les contenus numériques publiés sur le Web que l'on peut analyser en s'inspirant de certaines affaires relatives au droit d'auteur ou au droit à la vie privée.

3.5 Production des élèves

Essentiellement une production informatique : site fonctionnel sur un serveur Web.

3.6 Évaluation

Sur le mini-projet présenté devant un jury :

- investissement, pertinence des recherches
- production informatique : mobilisation des connaissances et des compétences, correction syntaxique (passage du validateur W3C), pertinence de la page Web et efficacité du message véhiculé.
- soutenance orale (construction de l'exposé, qualité de l'expression orale, richesse des connaissances mises en jeu, utilisation du vocabulaire technique approprié).

4 / Outils

- Un bon éditeur de texte : NotePad++ convient tout à fait (mais seulement sous Windows) (<http://notepad-plus-plus.org>), mais aussi Bluefish (<http://bluefish.openoffice.nl>).
- Module "Firebug" de développement Web pour Firefox : <https://addons.mozilla.org/fr/firefox/addon/firebug/>
- Validateurs HTML et CSS : <http://validator.w3.org>
- Logiciel client FTP : <http://filezilla-project.org/>
- Serveur Web : HTTP server Project Apache : <http://httpd.apache.org/>
Kit d'installation Apache : <http://www.apachefriends.org/fr/xampp.html>

5 / Auteur

Fabien Le Bris, professeur de STI, académie de Nantes

Créer des pages « Web »

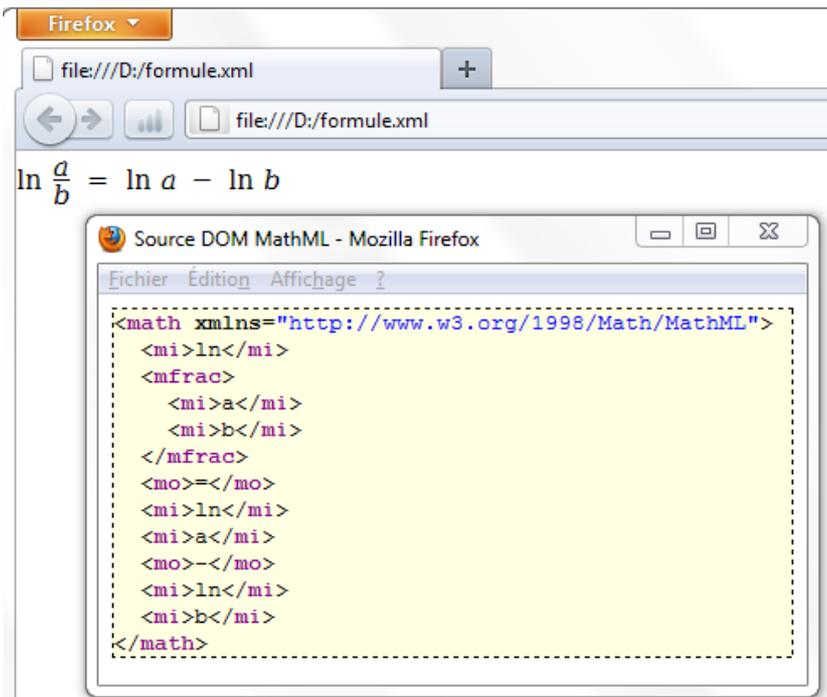
1 / Activités de création de pages Web

1.1 Introduction aux langages de balisage

La notion de page Web et son affichage pourront simplement être introduits en affichant le code source HTML d'une page Web simple à partir du navigateur. On introduit alors le principe des balises sémantiques que l'on trouve dans d'autres langages de description de contenus.

Parmi ces langages de balisage, on notera le langage XML qui est notamment sous-jacent aux formats Microsoft Office (depuis la version 2003) et OpenDocument.

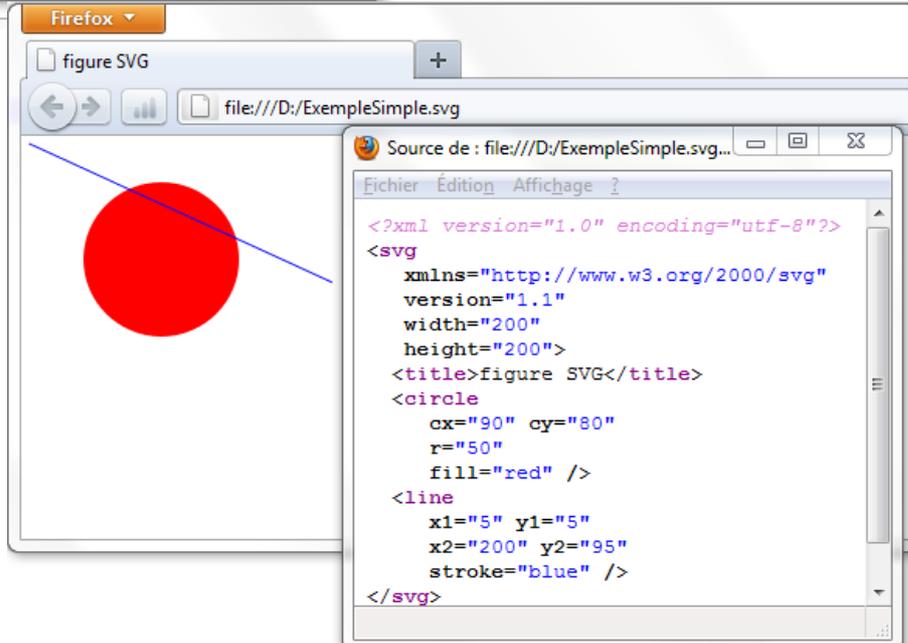
On pourra expérimenter avec les élèves quelques extensions du XML, recommandées par le W3C, et spécialisées dans la description de formules mathématiques ou le dessin vectoriel.



The screenshot shows a Firefox browser window with the address bar displaying 'file:///D:/formule.xml'. The main content area shows the mathematical formula $\ln \frac{a}{b} = \ln a - \ln b$. An 'Source DOM MathML - Mozilla Firefox' window is open, showing the following XML code:

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <mi>ln</mi>
  <mfraction>
    <mi>a</mi>
    <mi>b</mi>
  </mfraction>
  <mo>=</mo>
  <mi>ln</mi>
  <mi>a</mi>
  <mo>-</mo>
  <mi>ln</mi>
  <mi>b</mi>
</math>
```

MathML est un langage basé sur XML décrivant des formules mathématiques (formules que l'on rédige aisément à l'aide d'assistants au sein du logiciel OpenOffice.org ou de LibreOffice) ; on peut ensuite afficher ces formules dans des pages web qui seront correctement rendues par des navigateurs récents.



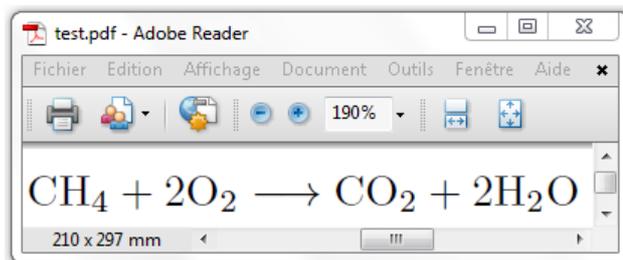
The screenshot shows a Firefox browser window with the address bar displaying 'file:///D:/ExempleSimple.svg'. The main content area shows a red circle with a blue line passing through it. An 'Source de : file:///D:/ExempleSimple.svg...' window is open, showing the following XML code:

```
<?xml version="1.0" encoding="utf-8"?>
<svg
  xmlns="http://www.w3.org/2000/svg"
  version="1.1"
  width="200"
  height="200">
  <title>figure SVG</title>
  <circle
    cx="90" cy="80"
    r="50"
    fill="red" />
  <line
    x1="5" y1="5"
    x2="200" y2="95"
    stroke="blue" />
</svg>
```

SVG est un langage de description de graphiques vectoriels (également basé sur XML), que l'on peut aussi insérer dans des pages web et voir au moyen d'un navigateur récent. Ces images peuvent être créées au moyen de logiciels spécialisés comme Inkscape.

On peut aussi montrer d'autres langages fonctionnant sur le même principe de **balises** comme *LaTeX*, particulièrement prisé pour la composition de documents scientifiques et techniques. *LaTeX* dispose de nombreux modules d'extensions dédiés à la physique, la chimie, l'électronique, etc...

```
\documentclass{article}
\begin{document}
\[
\mathrm{
CH_4+2O_2\longrightarrow CO_2+2H_2O
}
\]
\end{document}
```



1.2 Langages HTML/CSS

Les premières activités autour du HTML tourneront donc autour de la rédaction et/ou l'analyse de contenus purement textuels simples et utilisant les balises sémantiques de base. Les pages statiques seront rédigées grâce à un éditeur de texte (avec coloration syntaxique) et testées en local dans le navigateur. On souligne le respect de la syntaxe et des normes (ici dictées par le W3C) qui sont des compétences que l'on retrouve dans toutes activités de programmation.

Le développement peut se poursuivre avec l'intégration de contenus multimédia (images, animations flash,...) et on entrevoit les problèmes liés à la disposition des éléments dans la page.

La page précédente sera soumise à plusieurs mises en forme grâce à des **feuilles de style** (on pourra faire le parallèle en expérimentant les feuilles de style des documents texte Word Office ou Open/LibreOffice).

Il est important de mettre en évidence les avantages de la séparation du contenu de la page (fichier HTML) et de sa mise en forme par des feuilles de style (fichier CSS), notamment :

- différencier le travail de rédaction des contenus à l'aide de balises sémantiques et le travail de mise en forme ;
- réutilisation d'une même feuille de style pour différentes pages ;
- mise en forme des contenus destinés aux différents médias (écran d'ordinateur ou de mobile, imprimante, afficheur tactile braille, etc.) ;
- possibilités de doter les pages de présentations alternatives au choix de l'utilisateur et favoriser l'accessibilité de la page à tous les utilisateurs (par exemple à ceux qui ont des problèmes de vue).

On privilégiera une approche expérimentale pour découvrir le langage CSS à l'aide d'assistants de développement Web comme le module FireBug pour Firefox (voir photo de la page 1).

Sur une situation simple limitant la propagation des styles en cascade et l'utilisation des propriétés d'héritage, l'élève repère un élément particulier de la page dans le code HTML (une boîte de texte, une image, ...).

Il découvre l'imbrication des balises, le sélecteur CSS « Class » pour déclarer des styles personnalisés ou surcharger les styles standards.

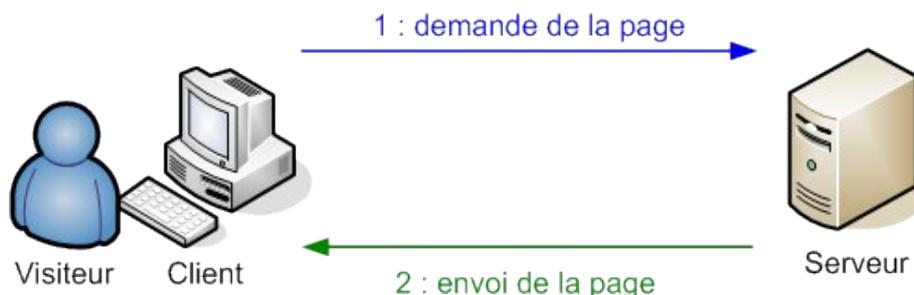
Dans la partie du code CSS correspondant, il modifie les paramètres (caractéristiques du texte, des polices ou propriétés de positionnement) et constate l'effet sur l'élément lui-même ou son organisation dans la page (positionnement absolu ou relatif).

Enfin, un site Web étant donc un ensemble de pages liées entre elles, on met en œuvre la navigation entre pages par des liens hypertextes ou à différents endroits d'une même page avec des ancres.

1.3 Le cheminement des pages sur le réseau Internet

L'élève devra finaliser un TP ou son projet en organisant ses fichiers puis en les téléchargeant grâce au protocole FTP sur un serveur Web.

Cette activité permet de rendre compte ensuite du fonctionnement du site « en production » et d'explicitier le cheminement à partir de la requête du client jusqu'au retour du code HTML dont le contenu est interprété par le navigateur (notion d'architecture « client/serveur », de serveur « Web », d'adressage Web, re-



1.4 Aspects sociétaux et juridiques

On sensibilisera l'élève à la réglementation sur les contenus numériques mis en ligne par l'intermédiaire d'un site Web (en particulier la réglementation française sur les droits d'auteur et droits à la vie privée) :

- Ai-je le droit de publier tel ou tel élément sur la vie privée de celui dont je parle ?
- Le texte que je veux publier est-il protégé par le droit d'auteur ?

Les questionnements associés aux droits à l'image seront abordés lors des séquences dédiées spécifiquement à l'image numérique:

- Qui est le propriétaire de l'image ? Ai-je le droit de la publier sur Internet ?
- L'image d'une personne peut-elle porter atteinte à la vie privée ?
- Quels sont les risques liés à l'utilisation illégale et irrégulière de l'image ?

L'amélioration de la qualité des sites favorisant la consultation des pages par tous les internautes, quelque soit leur contexte de navigation est un enjeu sociétal qui va au-delà du seul cas des personnes handicapées. Il est bien dans cette séquence d'évoquer au travers des activités menées autour du CSS, les efforts menés en faveur de l'accessibilité des sites.

2 / Mini-projet

Exemples de réalisations possibles en projet individuel : pages de présentation d'un produit, d'un service (du lycée). Page de contact, CV, etc.

À partir d'un cahier des charges, de « patrons » HTML/CSS, l'élève devra :

- compléter/modifier/personnaliser le « patron » HTML
- idem pour le « patron » CSS. Différentes feuilles de styles pourront être réalisées (utilisation d'assistants, de « templates » disponibles sur le Web) selon le contexte (affichage à l'écran ou au vidéo-projecteur, impression, amélioration de l'accessibilité de la page).

On fait remarquer aux élèves que la conception graphique de la page est un support fondamental du message que l'auteur du site souhaite transmettre. Pour véhiculer un message efficace il faut donc porter un grand soin au design de la page et on s'aidera des quelques principes érigés par les professionnels du Web Design (par exemple ceux désignés sous l'acronyme C.R.A.P²) afin d'apporter une véritable valeur en terme de communication.

- valider le respect des normes de codage à l'aide de validateurs HTML-CSS du W3C
- mettre le site « en production » sur un serveur Web (téléchargement FTP).

3 / Références

3.1 Standards du Web

- standards W3C - Web design and Applications : <http://www.w3.org/standards/webdesign>
- Web Design Group : <http://www.htmlhelp.com>
- Open Web Group : <http://openweb.eu.org>

3.2 Extensions basées sur XML

- MathML : <http://www.w3.org/Math/>
- Scalable Vector Graphics (SVG) : <http://www.w3.org/Graphics/SVG>

2 C.R.A.P : Contraste, Répétition, Alignement et Proximité

3.3 Accessibilité

- Pourquoi rendre un site accessible : <http://www.braillet.net.org/accessibilite/guide/access.htm>
- « Améliorez la qualité et l'accessibilité de vos sites Web » : <http://openweb.eu.org/articles/qualite-accessibilite>
- Les documents de référence de l'administration électronique – Accessibilité : <http://referenc.es.modernisation.gouv.fr>
- W3C : Web Accessibility Initiative : <http://www.w3.org/WAI/gettingstarted/Overview.html>

3.4 Aspects juridiques

- Guide du droit d'auteur : <http://www.educnet.education.fr/legamedia/guide/droit-auteur>
- La diffusion d'informations relatives à la vie privée : <http://www.educnet.education.fr/legamedia/fondamentaux/vie-privee/protection/droit-vie/b--diffusi/?searchterm=droit%20%C3%A0%20la%20vie%20priv%C3%A9e>
- « Droit à l'image et droit de l'image » : <http://www.cndp.fr/savoirscdi/index.php?id=870>
- Articles 226-1 à 226-8 du code civil : http://www.legifrance.gouv.fr/affichCode.do;jsessionid=8D548255C29F739203A1A063B65FFA6F.tpdjo03v_3?idSectionTA=LEGISCTA000006165309&cidTexte=LEGITEXT000006070719&dateTexte=20110619
- <http://www.legifrance.gouv.fr/affichCodeArticle.do?cidTexte=LEGITEXT000006070719&idArticle=LEGIARTI000006417938&dateTexte=20110619>

3.5 Langages HTML et CSS

- W3C-Feuilles de style en cascade : <http://www.w3.org/Style/CSS>
- W3C-Tutoriel CSS : débuter avec HTML + CSS : <http://www.w3.org/Style/Examples/011/firstcss.fr.html>
- Zen Garden – The Beauty of CSS Design : <http://www.csszengarden.com>

3.6 Web Design

Il est recommandé de lire sur le blog <http://www.presentationzen.com> l'article :

The Big Four : Contrast, Repetition, Alignment, Proximity,

publié à l'adresse : http://www.presentationzen.com/chapter6_spread.pdf

Annexe : Créer des pages « Web »

1 / But de l'activité

Créer une page Web en utilisant un « patron » HTML et CSS en s'aidant du module d'inspection de code *Firebug* pour le navigateur *Firefox*.

Firebug : <https://addons.mozilla.org/fr/firefox/addon/firebug>

On utilisera un éditeur de texte à coloration syntaxique comme *NotePad++* ou *Bluefish*.

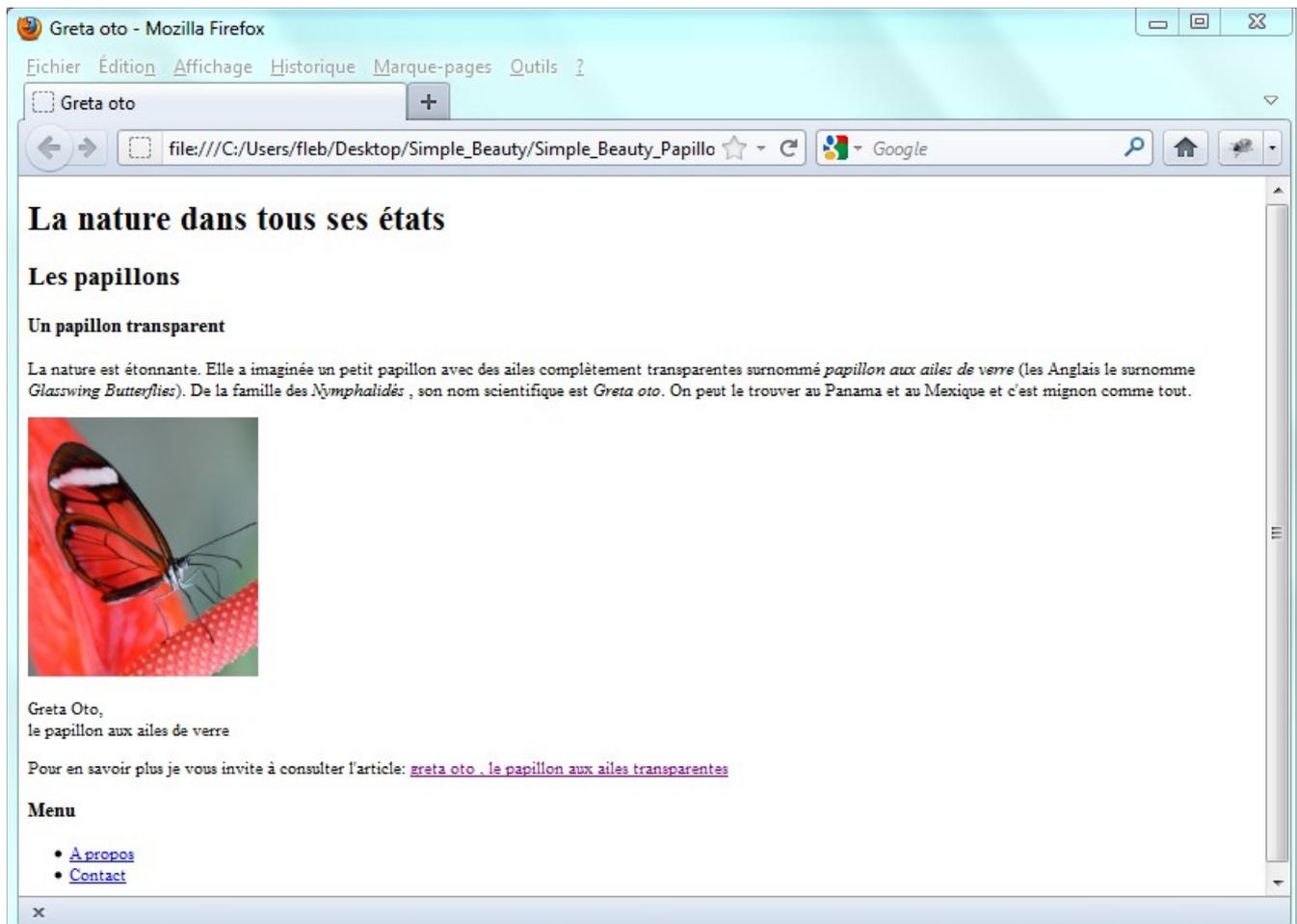
Notepad++ : <http://notepad-plus-plus.org/fr>

2 / Structuration de la page, langage HTML

Il s'agit dans un premier temps de rapidement mettre en activité les élèves sur la structuration d'une page en langage HTML.

L'idée fondamentale à faire passer étant de bien différencier ce qui relève de la **description de la page** (en indiquant la « fonction » de telle ou telle partie du texte grâce aux balises sémantiques : citation, passage important, titre, paragraphe...) et ce qui relève de sa **mise en forme** (espacement, police de caractères, couleurs, placement des images...).

L'exercice type peut consister à fournir une copie d'écran du résultat à obtenir, mais l'élève doit pouvoir aussi librement proposer sa page au contenu personnalisé si celle-ci reste raisonnable en complexité :



La structure du document est simple dans un premier temps, des titres et sous-titres, des paragraphes, des listes ordonnées ou non. Il comprendra au moins des liens hypertextes et des images.

On constate la (piètre) mise en forme par le navigateur.

3 / Mise en forme, les feuilles de style en langage CSS

Pour mettre en forme la page, on fournit des « patrons » (ou « templates ») HTML et CSS. On peut en trouver en utilisation libre sur Internet (attention toutefois aux conditions et licences d'utilisation).

Par exemple :

- Simple beauty : www.oswd.org/design/preview/id/3697
- Blue : templates.creakaz.com/blue/
- Touching Design : www.etomite.com/files/file/135-touching-design/
- ...



Ces patrons seront simples à analyser et à adapter. Ils comportent au moins deux fichiers séparés *index.html* et *style.css*.

Dans le fichier HTML, on fait remarquer la nécessaire liaison avec la feuille de style CSS dans l'en-tête, au

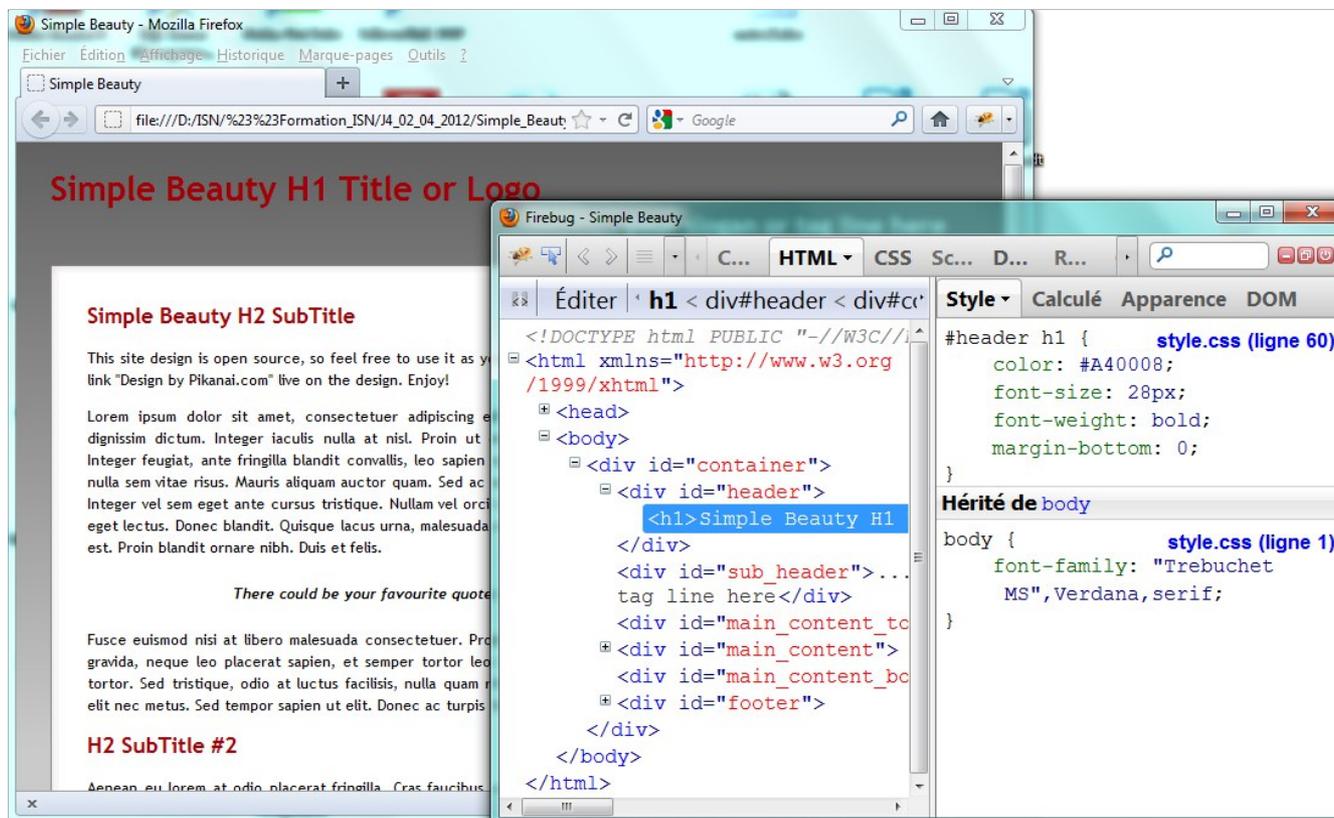
niveau de la ligne :

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

L'élève choisit un patron selon sa convenance et peut commencer le travail d'intégration de son contenu.

Auparavant il doit analyser la structuration des fichiers HTML et CSS du patron. L'outil proposé est le logiciel d'inspection *Firebug* pour le navigateur *Firefox*.

Par exemple avec le patron nommé « Simple Beauty » :



On découvre à l'intérieur du corps de la page (balise `<body>...</body>`), les structures personnalisées et imbriquées de façon hiérarchique. La structure « container » comprend des sous-structures « header », « main content », « footer » etc. La fenêtre *Firebug* est séparée en deux et permet de visualiser simultanément le code HTML et le style CSS correspondant à la sélection d'un élément HTML.

Par exemple, en sélectionnant l'élément `<h1> Simple Beauty H1 Title or Logo </h1>`, on découvre comment le style de l'élément `h1` a été surchargé dans la structure « Header », structure « Header » qui hérite elle-même du style de l'élément « body » en ce qui concerne la police de caractères.

On peut expérimenter, tester différents paramètres (ici la taille de la fonte en pixels) et visualiser directement l'effet de la modification sur la page (sans crainte de modifier le fichier CSS dans le mode d'inspection de *Firebug*).

```
#header h1 { style.css (ligne 60)
  color: #A40008;
  font-size: 28px;
  font-weight: bold;
  margin-bottom: 0;
}
```

Ci-dessous, la structure « Main-content » comprend deux sous-structures « Content » et « Menu » avec un style permettant l'affichage sur deux colonnes.

The image shows a Mozilla Firefox browser window displaying a website titled "Simple Beauty". The website layout includes a header with an H1 title and a slogan, a main content area with an H2 subtitle and a paragraph of Lorem Ipsum text, and a right-hand sidebar with a "Main menu" and "Sub menu". Red boxes and arrows highlight the "content" and "menu" areas, with callouts indicating their respective CSS classes.

Annotations on the right side of the browser window:

- class= "content"
- class= "menu"

The Firebug developer tool is open, showing the HTML structure of the page. The following code is visible in the "Éditer" (Edit) pane:

```

<div id="header">
  <h1>Simple Beauty H1 Title or Logo
</div>
<div id="sub_header">...
tag line here</div>
<div id="main_content">
  <div id="main_content">
    <div class="content">
    <div class="menu">
  </div>
  <div id="clear"></div>
</div>
<div id="main_content_bottom">
  <div id="friends">

```

The "Style" pane shows the CSS for the #main_content element:

```

#main_content {
  background: url("images/background.png") repeat-y scroll 0 0;
  #FFFFFF;
  margin: 0 auto;
  width: 100%;
}

```

The "Hérité de body" (Inherited from body) pane shows the font-family for the body element:

```

body {
  font-family: "Trebuchet MS", Verdana, serif;
}

```

En jouant avec les propriétés `float`, on découvre le principe des boîtes « flottantes » en inversant la disposition des deux colonnes.

The image illustrates the effect of the `float` property in CSS. It shows two browser windows side-by-side. The top window shows a page with a header, a main content area, and a right sidebar. The bottom window shows the same page but with the sidebar moved to the left, demonstrating how the `float` property can be used to rearrange the layout. A red arrow points from the top window to the bottom window, indicating the change. Below the browser windows is a screenshot of a web editor showing the HTML and CSS for the page. The CSS rule for the `.content` class is highlighted, showing `float: right`.

```
HTML
```

```
<div id="header">  
  <h1>Simple Beauty H1 Title or Logo  
</div>  
<div id="sub_header">...  
tag line here</div>  
<div id="main_content_top">  
<div id="main_content">  
  <div class="content">  
    <div class="menu">  
      <div id="clear"></div>  
    </div>  
  <div id="main_content_bottom">  
    <div id="friends">  
      <div id="clear"></div>  
    </div>  
  </div>  
</div>
```

```
Style
```

```
.content {  
  float: right;  
  font-size: 12px;  
  padding: 0 30px;  
  text-align: justify;  
  width: 510px;  
}
```

```
Hérité de body
```

```
body {  
  font-family: "Trebuchet  
  MS", Verdana, serif;  
}
```

En sélectionnant l'élément correspondant à la boîte « menu » dans la partie HTML, on découvre notamment que celle-ci « flotte » à droite dans le conteneur de niveau supérieur « main-content », que sa largeur est de 139 pixels, puis on découvre la signification de `padding: 0 20px` (voir ci-dessous) et le système des marges CSS en expérimentant des valeurs de 5 ou 10 pixels.

Visualisation des marges de 20 pixels

```

<div id="main_content_top">
<div id="main_content">
  <div class="content">
    <div class="menu">
      <div id="clear"></div>
    </div>
  </div>
<div id="main_content_bot">
<div id="footer">

```

```

.menu {
  border-left: 1px solid #8C8484;
  float: right;
  font-size: 12px;
  padding: 0 20px;
  width: 139px;
}

```

On peut également découvrir plus en détail le système de marges en expérimentant l'onglet *Apparence* de *Firebug*.



margin 0

border 0

padding 0

0 1 5 139 x 259 5 0 0

0

0

0

position: static

z: auto

box-sizing: content-box

Au final, il reste à intégrer le contenu HTML personnalisé dans le patron.

La nature dans tous ses états

Les papillons

Un papillon transparent

La nature est étonnante. Elle a imaginé un petit papillon avec des ailes complètement transparentes surnommé *papillon aux ailes de verre* (les Anglais le surnomme *Glasswing Butterflies*). De la famille des *Nymphalidés*, son nom scientifique est *Greta oto*. On peut le trouver au Panama et au Mexique et c'est mignon comme tout.



*Greta Oto,
le papillon aux ailes de verre*

Pour en savoir plus je vous invite à consulter l'article: [greta oto , le papillon aux ailes transparentes](#)

Menu

- [A propos](#)
- [Contact](#)



Copyright © 2012 | [Mon Site](#) | Design by [Pikanai.com](#)

Le résultat n'est pas immédiatement satisfaisant, il reste des paramètres à adapter (couleurs, polices de caractères, taille et disposition des éléments etc ...) mais l'élève peut expérimenter sa nouvelle mise en forme avec *Firebug*.

Si l'élève a compris le principe des boîtes flottantes, on peut le diriger et l'aider à compléter le CSS en créant une nouvelle structure simple pour disposer le paragraphe et la photo sur deux colonnes, par exemple :

La nature dans tous ses états

Les papillons

Un papillon transparent

La nature est étonnante. Elle a imaginé un petit papillon avec des ailes complètement transparentes surnommé *papillon aux ailes de verre* (les Anglais le surnomme *Glasswing Butterflies*). De la famille des *Nymphalidés*, son nom scientifique est *Greta oto*. On peut le trouver au Panama et au Mexique et c'est mignon comme tout.



*Greta Oto,
le papillon aux ailes de verre*

Pour en savoir plus je vous invite à consulter l'article: [greta oto , le papillon aux ailes transparentes](#)

Menu

- [A propos](#)
- [Contact](#)



Copyright © 2012 | [Mon Site](#) | Design by [Pikanai.com](#)

4 / Validation W3C

À tout moment dans le développement, l'élève devra veiller à ce que son code HTML et CSS reste conforme aux recommandations du W3C (World Wide Web Consortium www.w3.org).

Il utilisera à cet effet les outils de validations officiels du W3C (<http://validator.w3.org>), qui en plus de signaler les erreurs de conformité, assiste le développeur en décrivant l'erreur constaté et en faisant même des propositions de correction.

The screenshot shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are navigation links: "Jump To: Notes and Potential Issues Validation Output". A red banner indicates "Errors found while checking this document as HTML 4.01 Transitional!". Below this, a table shows the result: "Result: 4 Errors, 4 warning(s)". There is a file selection box with a "Parcourir..." button. Below the file selection box, there is another blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below this header, there are navigation links: "Jump To: Congratulations Icons". A green banner indicates "This document was successfully checked as HTML 4.01 Transitional!". Below this, a table shows the result: "Result: Passed". There is a file selection box with a "Parcourir..." button. Below the file selection box, there is a note: "Use the file selection box above if you wish to re-validate the uploaded file index.html". Below the note, there are two rows of configuration options: "Encoding: iso-8859-1 (detect automatically)" and "Doctype: HTML 4.01 Transitional (detect automatically)".

Résultats de la validation W3C CSS de TextArea (CSS niveau 3)

Félicitations ! Aucune erreur trouvée.

Ce document est valide conformément à la recommandation [CSS niveau 3](#) !

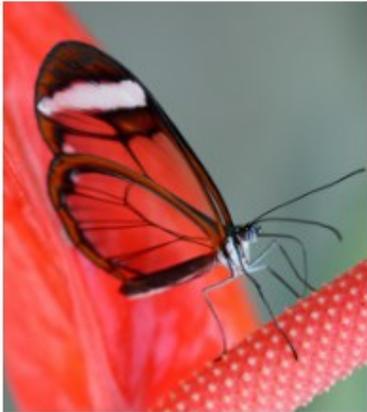
Le but est alors d'obtenir le sésame du W3C ...

La nature dans tous ses états

Les papillons

Un papillon transparent

La nature est étonnante. Elle a imaginé un petit papillon avec des ailes complètement transparentes surnommé *papillon aux ailes de verre* (les Anglais le surnomme *Glasswing Butterflies*). De la famille des *Nymphalidés*, son nom scientifique est *Greta oto*. On peut le trouver au Panama et au Mexique et c'est mignon comme tout.



Greta Oto,
le papillon aux ailes de verre

Pour en savoir plus je vous invite à consulter l'article: [greta oto , le papillon aux ailes transparentes](#)

Menu

- [A propos](#)
- [Contact](#)



Copyright © 2012 | [Mon Site](#) | Design by [Pikanai.com](#)



5 / Une autre idée : créer un CV en ligne

Le site <http://templates.creakaz.com/CV> propose des patrons HTML et CSS gratuits pour la publication d'un *Curriculum Vitae* sur Internet.

Un travail de rédaction de CV avec un autre enseignant à des fins de publication en ligne peut être une source de motivation supplémentaire pour certains élèves dans le cadre d'un mini-projet.

Exemple de travail d'analyse et d'inspection du code avec le module *Firebug* :

OBJECTIFS

Mes objectifs
Mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs
mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs

EXPERIENCE PROFESSIONNELLE

20.07.06 - 01.10.06 Mes expériences professionnelles Mes expériences Site de la Société

Firebug - CV

```
Éditer 'div#top_cont <div#c'  
#top_cont {  
  float: left;  
  width: 200px;  
}  
* {  
  margin: 0;  
  padding: 0;  
}  
Hérité de body  
body, td {  
  color: #9099AE;
```

OBJECTIFS

Mes objectifs
Mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs
mes objectifs mes objectifs mes objectifs mes objectifs mes objectifs

EXPERIENCE PROFESSIONNELLE

20.07.06 - 01.10.06 Mes expériences professionnelles Mes expériences Site de la Société

Firebug - CV

```
Éditer 'div#top_cont <div#c'  
#top_cont {  
  float: right;  
  width: 200px;  
}  
* {  
  margin: 0;  
  padding: 0;  
}  
Hérité de body  
body, td {  
  color: #9099AE;
```

Sandy Dupont

123, rue des Palmiers
75001 Marseille
Suisse

06.06.06.06.06
02.38.01.01.01

mail
www.blog.tld