

TP1 : XHTML et CSS (Corrigé)

Nous allons utiliser pour tous les exercices (de ce tp et des suivants) le serveur de l'IUT qui s'appelle aquanux. Il permet de mettre en ligne tous les documents que vous souhaitez. Pour cela, il vous suffit de mettre ces documents dans le répertoire `public_html` qui se trouve dans votre répertoire `home`. Pour accéder à ces documents à travers un navigateur, il faut alors taper l'adresse Web :

`http://aquanux/~VOTRE_NUMERO_ETUDIANT/chemin/nom_du_fichier`

où `chemin` correspond au chemin où se trouve votre fichier à partir du répertoire `public_html`. Par exemple, si votre numéro d'étudiant est le 001, si vous créez un dossier TP1 dans votre répertoire `public_html` et un fichier `exercice1.html` dans TP1, pour accéder à ce fichier, vous devez spécifier l'adresse Web :

`http://aquanux/~001/TP1/exercice1.html`

Comme le navigateur sait interpréter les codes XHTML et CSS, il est possible d'afficher une page en local, c'est-à-dire sans passer par le Web. Il suffit pour cela d'ouvrir un fichier avec le navigateur (le protocole utilisé n'est alors plus `http` mais `file`). Cependant, pour tous les tps, nous ouvrirons les documents de la manière décrite précédemment car, par la suite, nous utiliserons le langage PHP que le navigateur est incapable d'interpréter. Il sera donc impératif d'ouvrir les fichiers en passant par le serveur.

Pour tous les exercices, nous utiliserons le navigateur Mozilla Firefox.

Remarque : Dans tous les TPs, les exercices notés par une astérisque ont été repris du cours réalisé par Marcel BOSCH, enseignant à l'IUT de Villetaneuse.

Exercice 1 : Première page Web

Écrire une première page en XHTML contenant un titre de niveau 1 et un paragraphe. Vérifier que cette page est valide grâce au validateur <http://www.validome.org/>. L'afficher avec le navigateur Firefox. Vérifier que le code source de la page est bien celui qui a été écrit. Ajouter ensuite une liste ordonnée contenant 3 items et une liste non ordonnée contenant 2 items. (Vérifier systématiquement que le code est valide grâce au validateur). Ajouter ensuite le tableau suivant

Nom de la balise	Utilisée pour définir	balise auto-fermante
p	un paragraphe	non
img	une image	oui
ol	une liste ordonnée	non
ul	une liste non-ordonnée	non
li	un item d'une liste	non
h1	un titre de niveau 1	non
a	un lien hypertexte	non
br	un saut de ligne	oui

Correction :

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
4 <head>
5   <title>Un essai</title>
6   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
7 </head>
```

```

8 <body>
9   <h1> Ceci est mon premier titre </h1>
10  <p> et ceci mon premier paragraphe </p>
11  <ol>
12      <li> point numéro 1 </li>
13      <li> point numéro 2 </li>
14      <li> point numéro 3 </li>
15  </ol>
16
17  <ul>
18      <li> Une chose </li>
19      <li> Encore une chose </li>
20  </ul>
21  <table>
22      <tr>
23          <th> Nom de la balise </th>
24          <th> Utilisé pour définir </th>
25          <th> balise auto-fermante </th>
26      </tr>
27      <tr><td> p </td><td> un paragraphe </td><td>non</td> </tr>
28      <tr><td> img </td><td> une image </td><td>oui</td> </tr>
29      <tr><td> ol </td><td> une liste ordonnée </td><td>non</td> </tr>
30      <tr><td> ul </td><td> une liste non-ordonnée </td><td>non</td> </tr>
31      <tr><td> li </td><td> un item d'une liste </td><td>non</td> </tr>
32      <tr><td> h1 </td><td> un titre de niveau 1 </td><td>non</td> </tr>
33      <tr><td> a </td><td> un lien hypertexte </td><td>non</td> </tr>
34      <tr><td> br </td><td> un saut de ligne </td><td>oui</td> </tr>
35  </table>
36 </body>
37 </html>

```

Exercice 2 : Comprendre l'interaction client/serveur*

Nous allons réellement voir quels sont les messages envoyés par le client et par le serveur. Pour cela, nous allons utiliser le logiciel netcat (ce logiciel se lance grâce à la commande `nc`) qui permet :

- en mode client : de se connecter à un serveur et dialoguer avec lui,
- en mode serveur : de créer un "serveur", c'est-à-dire, d'écouter sur le réseau en attendant qu'un client se connecte, puis dialoguer avec lui.

Remarque : Pour communiquer sur un réseau les programmes doivent spécifier des "ports" qui sont identifiés par des numéros. Par exemple, le port pour le Web (protocole `http`) est normalement le port numéro 80. Donc, quand on tape l'adresse Web `http://exemple.org/fichier.html`, le navigateur se connecte au port 80 du serveur `exemple.org`. Pour spécifier le port à utiliser dans une adresse, il suffit de rajouter le numéro du port après le nom du domaine (ou l'adresse IP) en séparant les deux par le symbole ":" - par exemple `http://exemple.org:80/fichier.html`.

Question 2.1 : Pour comprendre comment fonctionne le logiciel netcat, nous allons commencer par créer un serveur et un client dans deux terminaux différents et établir un dialogue entre les deux.

1. Ouvrir deux terminaux et les mettre côte à côte (pour pouvoir les voir en même temps)
2. Créer un serveur dans un terminal à l'aide de la commande `nc -l -p 1234`
Explication : `-l` signifie "listen" (écouter) et `-p 1234` indique qu'il faut utiliser le port 1234. (On aurait pu prendre n'importe quel port supérieur à 1024.) Attention, sur certaines distributions linux, il faut supprimer les caractères `-p` de la commande.
3. Dans l'autre terminal, se connecter au serveur en lançant un client grâce à la commande `nc localhost 1234`

Explication : `localhost` est le nom de la machine sur laquelle vous êtes.

4. Taper du texte suivi d'entrée dans les deux terminaux, normalement il devrait s'afficher sur l'autre.

Voilà, vous avez établi un dialogue client/serveur ! Vous pouvez terminer le dialogue avec le raccourci `ctrl-D` qui signifie "fin de fichier". (Si `ctrl-D` ne fonctionne pas, utilisez `ctrl-C`.)

Question 2.2 : netcat en serveur, Firefox en client

Dans l'exercice précédent, deux `nc` ont dialogué entre eux. Maintenant, nous allons faire dialoguer un "`nc`" en serveur, et le client sera cette fois le navigateur (Firefox).

1. Ouvrir un terminal et une fenêtre firefox et les placer côte à côte (pour pouvoir voir les deux en même temps),
2. créer un serveur dans un terminal grâce à la commande `nc -l -p 1234`,
3. Dans firefox, entrer l'adresse suivante : `http://localhost:1234/monfichier.html`

Regardez bien ce qui s'affiche. Firefox vient de se connecter à votre serveur ! En regardant votre cours, vous devriez pouvoir comprendre ce que vous dit Firefox. Vous pouvez lui répondre en tapant quelques lignes de texte. Puis finissez avec le raccourci `ctrl-D`. Observez bien ce dialogue. Le protocole `http`, c'est ça : un simple dialogue où le client et le serveur s'envoient du texte, rien d'autre.

Firefox vient de vous dire "Bonjour je voudrais la page `/monfichier.html`, je suis Firefox version xyz, j'accepte telle langue, tel encodage..."

Répondre à la requête en écrivant du code XHTML. Que faut-il écrire au début pour que Firefox comprenne que le texte renvoyé par le serveur est du code XHTML ?

Correction :

```

1 HTTP/1.1 200 OK
2 Content-Type: text/html
3
4 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
5                               "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
6 <!-- Ajouter tout le code XHTML -->
```

Question 2.3 : netcat en client, sur un serveur web

Maintenant on va utiliser "`nc`" en mode client pour se connecter et accéder à la page `http://www-lipn.univ-paris13.fr/~lacroix/Documents/IntroWeb/interaction.html`.

1. Comment utiliser netcat pour qu'il se connecte au bon site ?

Correction :

```
nc www-lipn.univ-paris13.fr 80
```

2. Quel message netcat doit-il envoyer au serveur pour recevoir la page Web ?

Correction :

```

1 GET /~lacroix/Documents/IntroWeb/interaction.html HTTP/1.1
2 Host: www-lipn.univ-paris13.fr
```

3. Le texte reçu correspond-t-il à la page Web demandée ?

Correction :

Oui. Pour s'en assurer, il suffit d'accéder à la page avec Firefox et de vérifier que le code source de la page est identique à celui renvoyé à netcat.

Exercice 3 : Utilisation de l'extension Firebug : onglet Réseau*

Nous allons apprendre à utiliser une partie de l'extension Firebug. Le lancement de cette extension se fait en cliquant sur l'insecte (bug en anglais) en bas à droite de l'écran. L'ouverture de Firebug fait apparaître une fenêtre contenant plusieurs onglets. Nous allons apprendre à utiliser l'onglet Réseau.

Accéder au site Web www.fsf.org. Ouvrir Firebug, cliquer sur l'onglet Réseau et recharger la page Web (raccourci F5). Répondre aux questions suivantes :

- Combien de requêtes sont faites pour afficher cette page ?
- À quel type de fichier correspond la majorité des requêtes ?
- Quelle est la taille totale de toutes les données téléchargées ?
- Quel est le nom du plus gros fichier téléchargé ?
- Quels sont les deux serveurs sur lesquels sont téléchargés les fichiers ?
- Quelle est la durée de téléchargement des fichiers nécessaires à l'affichage de la page ?

Correction :

- 27 requêtes
- des fichiers images (png, gif)
- 150.6 ko
- [deserve.png](http://www.fsf.org)
- <http://www.fsf.org/>, <http://static.fsf.org/>
- 3.07 secondes



Exercice 4 : Liens hypertextes

Nous allons travailler sur les liens hypertextes. Pour cela créer dans le répertoire `public_html` un répertoire `TP1bis`. Ajouter un fichier `page2.html` dans `TP1` et un fichier `page3.html` dans `TP1bis`, ces deux fichiers XHTML valides ne contenant qu'un titre (différent !). Dans le fichier du premier exercice, créer ensuite deux liens hypertextes sur les fichiers `page2.html` et `page3.html`. Ajouter également un lien hypertexte sur le site Web de l'IUT. Vérifier que tous les fichiers sont valides !

Correction :

Dans le fichier de l'exercice 1, rajouter :

```
1 <p>
2 <a href="page2.html"> Voici un lien permettant d'accéder au fichier page2.html </a> <br />
3 <a href="../TP1bis/page3.html"> Voici un lien permettant d'accéder au fichier page3.html </a>
4 <br /> <a href="www.iutv.univ-paris13.fr/"> Voici un lien sur l'IUT de Villetaneuse </a>
5 </p>
```



Exercice 5 : Arborescence d'un document XHTML

Donner l'arborescence du code XHTML suivant :

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
4 <head>
5 <title>Fichier utilisé pour déterminer l'arbre (ou arborescence) d'un document XHTML </title>
6 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
7 </head>
8 <body>
9 <h1> Un titre <strong> de niveau 1 </strong> </h1>
10 <p> Il est vraiment très important <strong> en <a href="http://fr.wikipedia.org/wiki/Xhtml">
11 XHTML </a> </strong> d'être capable de dessiner l'arbre ou arborescence d'un document XHTML.
12 Cela permet : </p>
13 <ol>
```

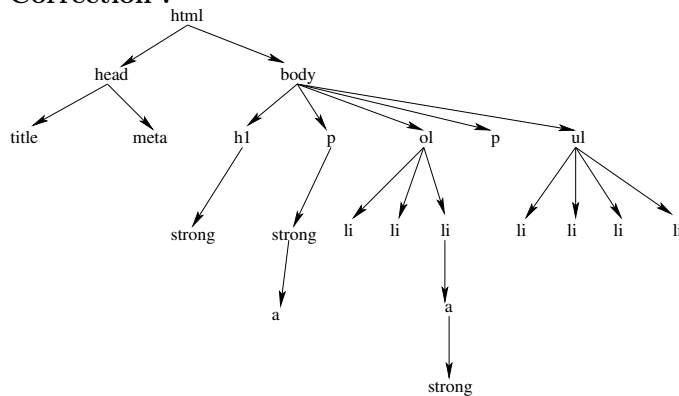
```

14     <li> de mieux comprendre le document que l'on crée </li>
15     <li> aux programmes de modifier le document (hors de ce cours) </li>
16     <li> d'utiliser correctement les
17     <a href="http://fr.wikipedia.org/wiki/CSS"> <strong> CSS </strong> </a> </li>
18 </ol>
19 <p> Il faut être capable de donner, pour chaque balise : </p>
20 <ul>
21     <li> ses descendants, </li>
22     <li> ses ascendants, </li>
23     <li> ses fils </li>
24     <li> son père </li>
25 </ul>
26 </body>
27 </html>

```

Quels sont les descendants et ascendants de la balise ol ? Quels sont ses enfants et parents ?

Correction :



◇

Exercice 6 : Utilisation des CSS

Le but de cet exercice est de créer un fichier CSS, appelé "testStyle.css", afin de modifier la page Web de l'exercice 1. Modifier l'apparence de la page Web selon les critères suivants :

- Ajouter une couleur de fond (jaune par exemple)
- Mettre les titres de niveau 1 en rouge et en italique
- Justifier les paragraphes
- Supprimer les puces dans les listes non ordonnées
- Mettre les items des listes ordonnées et non ordonnées en gris

Modifier également le fichier XHTML de l'exercice précédent grâce à ce fichier CSS. Ajouter également les modifications suivantes :

- Modifier la couleur des liens en vert
- les textes mis en valeur grâce à la balise **strong** sont affichés en rouge
- les textes dans les listes (ordonnées ou non ordonnées) mis en valeur grâce à la balise strong sont affichés en italique, en gras et en bleu (les autres restant affichés en rouge)

Vérifier que le fichier CSS est valide grâce au validateur (par exemple : <http://jigsaw.w3.org/css-validator/validator>)!

Correction :

Fichier testStyle.css :

```

1 h1
2 {
3     color : red;
4     font-style : italic ;
5 }

```

```
6
7 body
8 {
9     background-color: yellow;
10 }
11
12 p
13 {
14     text-align : justify;
15 }
16
17 ul, ol
18 {
19     color : gray;
20 }
21
22 ul
23 {
24     list-style-type: none;
25 }
26
27 a
28 {
29     color : green;
30 }
31
32 strong
33 {
34     color : red;
35 }
36
37 li strong
38 {
39     color : blue;
40     font-weight : bold;
41     font-style : italic;
42 }
```

Rajouter dans les fichiers des exercices 1 et 4 la ligne de code suivante (à l'intérieur de la balise head)

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

Exercice 7 : Bouton cliquable*

Créer un lien vers le site de l'IUT sous forme d'un bouton comme indiqué dans la figure suivante.



La couleur de fond est la couleur #aaa et celle des bords la couleur #555. Faire en sorte que lors du clic sur le lien, les couleurs soient inversées.

Correction :

```
1 <body>
2 <p> <a href="http://www.iutv.univ-paris13.fr/iutv/" class="bouton"> IUT </a> </p>
3 </body>
```

```

1  .bouton
2  {
3      background-color: #aaa;
4      color : white;
5      text-decoration : none;
6      border: 1px solid #999;
7      border-bottom: 2px solid #555;
8      border-right : 2px solid #555;
9      padding: .3em;
10     font-weight: bold;
11     font-family : sans-serif ;
12 }
13
14 .bouton:active
15 {
16     background-color: #555;
17     border-bottom: 2px solid #aaa;
18     border-right : 2px solid #aaa;
19 }
20 }
```

Exercice 8 : Priorité et Héritage en CSS

Considérons le code XHTML suivant

```

1  <body>
2  <p> a <strong> b <a href="../premierFichierHtml.html"> c </a> d </strong> e
3  <strong class="a"> f <a href="../premierFichierHtml.html" class="a"> g </a> h
4  </strong> i <strong class="a"> j <a href="../premierFichierHtml.html"> k
5  <em class="a"> l </em> m <em> n </em> o </a> p </strong> q </p>
6  </body>
```

et le code CSS associé

```

1  strong
2  {
3      color : yellow;
4  }
5
6  p
7  {
8      color : red;
9  }
10
11 strong.a
12 {
13     color : blue;
14 }
15
16 strong .a
17 {
18     color : purple;
19 }
20
21
22 strong > .a
23 {
```

```
24     color : silver ;
25 }
26
27 strong a
28 {
29     color : olive ;
30 }
```

Question 8.1 : Pour chaque règle, indiquer quelle est la cible du sélecteur.

Question 8.2 : Donner la priorité de chaque règle. Que se passe-t-il si l'on inverse les règles 4 et 5 dans le code CSS ?

Question 8.3 : Pour chaque lettre, indiquer quelle(s) règle(s) s'applique(nt).

Question 8.4 : Indiquer la couleur de chaque lettre.

Correction :

Cet exercice est à faire sous forme de TD. Leur demander d'écrire la représentation du code XHTML sous forme d'arbre (rajouter en pointillés les lettres). Leur dire que cet exercice (ou une variante) risque très fortement d'être dans un partiel.

Réponses aux questions 1 et 2 :

- **strong** : le texte fortement mis en valeur par une balise **strong** (001)
- **p** : les paragraphes (001)
- **strong.a** : le texte fortement mis en valeur par une balise **strong** appartenant à la classe "a" (011)
- **strong .a** : un élément appartenant à la classe "a" descendant de **strong** (011)
- **strong > .a** : un élément appartenant à la classe "a" enfant de **strong** (011)
- **strong a** : un lien hypertexte descendant de **strong** (002)

Si l'on inverse les règles 4 et 5, la règle 5 ne sert plus à rien. En effet, la règle 4 devient prioritaire (puisqu'écrite après la règle 5). De plus, toute cible de la règle 5 est aussi cible de la règle 4.

Réponses aux questions 3 et 4 :

- a : Règle 2 => couleur rouge
- b : Règle 1 => couleur jaune
- c : Règle 6 => couleur olive
- d : Règle 1 => couleur jaune
- e : Règle 2 => couleur rouge
- f : Règles 1 et 3 => couleur bleu
- g : Règles 4, 5 et 6 => couleur argent
- h : Règles 1 et 3 => couleur bleu
- i : Règle 2 => couleur rouge
- j : Règles 1 et 3 => couleur bleu
- k : Règle 6 => couleur olive
- l : Règle 4 => couleur violet
- m : Règle 6 => couleur olive
- n : Aucune règle => couleur donnée par le parent (p > **strong.a** > a) : olive
- o : Règle 6 => couleur olive
- p : Règles 1 et 3 => couleur bleu
- q : Règle 2 => couleur rouge

◇

Exercice 9 : Comprendre le positionnement des éléments

Télécharger les fichiers XHTML et CSS situés à l'adresse
<http://www-lipn.univ-paris13.fr/~lacroix/Documents/TP/IntroWeb/Positionnement/>.

Tester les cas suivants afin de bien comprendre l'utilisation de la propriété **position**.

Question 9.1 : Construire l'arbre de représentation du code XHTML du fichier téléchargé.

Question 9.2 : Dans le fichier CSS, modifier les valeurs **right** et **bottom** à 50% puis 25%.

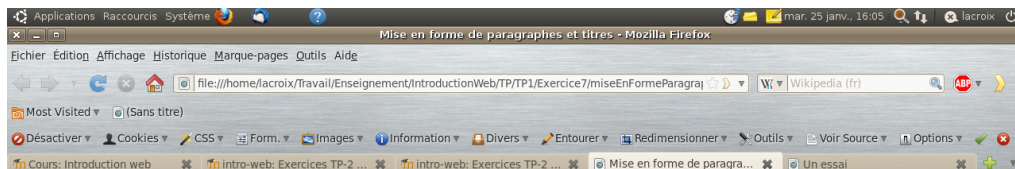
Question 9.3 : Remplacer ensuite **right** et **bottom** par **left** et **top** (tester 25%, 50% et 0).

Question 9.4 : Ajouter la propriété **position:relative**; pour le **div** et recommencer les tests des positions.

Question 9.5 : Déplacer la propriété `position:relative;` dans la balise `body` et recommencer encore une fois les tests.

Exercice 10 :

Question 10.1 : Créer un fichier XHTML valide composé de deux tableaux, de deux titres de niveau 1, d'un titre de niveau 3 et d'un paragraphe. Le rendu avec Firefox doit alors être celui donné par la figure 2.5.



Tableaux :

Col 1	Col 2	Col 3
Case 1	Case 2	Case 3
Case 4	Case 5	Case 6
Case 7	Case 8	Case 9
Case 10	Case 11	Case 12

Col 1	Col 2	Col 3
Case 1	Case 2	Case 3
Case 4	Case 5	Case 6
Case 7	Case 8	Case 9
Case 10	Case 11	Case 12

Paragraphe :

Titre du paragraphe

Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte. Du texte.



FIGURE 2.5 – Rendu sans CSS

Question 10.2 : Modifier, à l'aide d'un fichier CSS, l'apparence du titre de niveau 3 et du paragraphe pour obtenir le résultat de la figure 2.6. Afin de répondre à cette question, télécharger

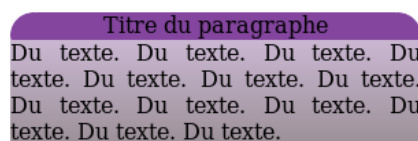


FIGURE 2.6 – Apparence du paragraphe avec le fichier CSS

les deux images nécessaires :

- <http://www-lipn.univ-paris13.fr/~lacroix/Documents/TP/IntroWeb/title.png>
- <http://www-lipn.univ-paris13.fr/~lacroix/Documents/TP/IntroWeb/degrade.png>

Correction :

```

1 h3
2 {
3     background-image : url('title.png');
4     background-repeat : no-repeat;
5     font-weight : normal;
6     font-size : medium;

```

```

7      width : 300px;
8      height : 20px;
9      text-align : center;
10     margin-bottom : 0px;
11     padding-bottom : 0px;
12 }
13
14 p
15 {
16     width : 300px;
17     margin-top : 0px;
18     padding-top : 0px;
19     text-align : justify ;
20     background-image : url('degrade.png');
21 }

```

Question 10.3 : Modifier les fichiers XHTML et CSS afin de mettre en page les tableaux comme sur la figure 2.7.

Col 1	Col 2	Col 3
Case 1	Case 2	Case 3
Case 4	Case 5	Case 6
Case 7	Case 8	Case 9
Case 10	Case 11	Case 12

FIGURE 2.7 – Apparence d'un tableau

Correction :

Code XHTML :

```

1      <table class="droite">
2      <tr>
3          <th> Col 1 </th> <th> Col 2 </th> <th> Col 3 </th>
4      </tr>
5      <tr class="impair">
6          <td> Case 1 </td><td> Case 2 </td><td> Case 3 </td>
7      </tr>
8      <tr class="pair">
9          <td> Case 4 </td><td> Case 5 </td><td> Case 6 </td>
10     </tr>
11     <tr class="impair">
12         <td> Case 7 </td><td> Case 8 </td><td> Case 9 </td>
13     </tr>
14     <tr class="pair">
15         <td> Case 10 </td><td> Case 11 </td><td> Case 12 </td>
16     </tr>
17     </table>

```

Code CSS :

```

1  table
2  {
3      width : 400px;
4      margin-left : auto;

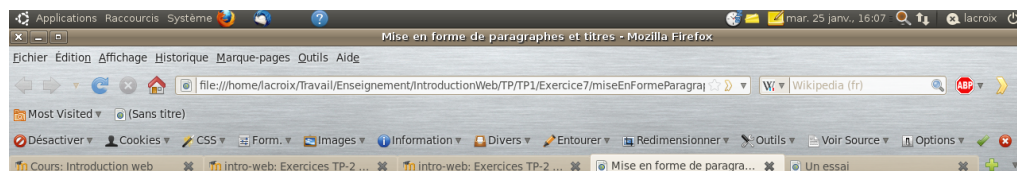
```

```

5     margin-right : auto;
6     border : 2px solid black;
7     border-collapse : collapse;
8     text-align : center;
9 }
10 th,td
11 {
12     border : 1px solid black;
13     width : 30%;
14     height : 40px;
15 }
16 tr { background-color : #DC2F2F; }
17
18 tr.impair { background-color : #F16D6D; }
19
20 tr.pair { background-color : #CB4040;}

```

Question 10.4 : Modifier les fichiers XHTML et CSS pour obtenir le résultat de la figure 2.8.



Tableaux :

Col 1	Col 2	Col 3
Case 1	Case 2	Case 3
Case 4	Case 5	Case 6
Case 7	Case 8	Case 9
Case 10	Case 11	Case 12

Col 1	Col 2	Col 3
Case 1	Case 2	Case 3
Case 4	Case 5	Case 6
Case 7	Case 8	Case 9
Case 10	Case 11	Case 12

Paragraphe :

Titre du paragraphe
 Du texte. Du texte. Du texte. Du texte.
 Du texte. Du texte. Du texte. Du texte.
 Du texte. Du texte. Du texte. Du texte.
 Du texte. Du texte. Du texte.



FIGURE 2.8 – Rendu avec CSS

Correction :

Modifier le XHTML comme suit :

```

1  ...
2      <h1> Tableaux : </h1>
3
4      <div class="mepTableaux">
5          <table class="gauche">
6              CODE DU TABLEAU
7          </table>
8
9          <table class="droite">
10             CODE DU TABLEAU
11         </table>

```

```
12     </div>
13
14     <h1> Paragraphe : </h1>
15     <h3> Titre du paragraphe </h3>
16     <p> TEXTE DU PARAGRAPHE </p>
17     ...
```

Code CSS :

```
1  .mepTableaux
2  {
3      width : 820px;
4      margin-left : auto;
5      margin-right : auto;
6      position : relative ;
7      height : 240px;
8  }
9  .gauche
10 {
11     position : absolute;
12     top : 0px;
13     left : 0px;
14 }
15 .droite
16 {
17     position : absolute;
18     bottom : 0px;
19     right : 0px;
20 }
```

Sans utiliser les classes "gauche" et "droite" et en ayant un seul tableau en **position:absolute**;

```
1  .mepTableaux table: first - child
2  {
3      margin-top : 0;
4      position : absolute;
5      top : 0px;
6      left : 0px;
7  }
8
9  .mepTableaux table
10 {
11     margin-top : 40px;
12     margin-right : 0;
13 }
14
15
16 .mepTableaux
17 {
18     width : 820px;
19     margin-left : auto;
20     margin-right : auto;
21     position : relative ;
22     height : 240px;
23 }
```

Exercice 11 : Modification du type de balise*

Créer un fichier CSS permettant d'obtenir le menu donné par la figure



à partir du code XHTML suivant

```

1 <ul class="menu">
2   <li><a href="accueil.html">Accueil</a></li>
3   <li class="active"><a href="infos.html" >Infos</a></li>
4   <li><a href="contact.html">Contact</a></li>
5   <li><a href="login.html">Se connecter</a></li>
6 </ul>

```

Pour cela, on pourra utiliser les propriétés :

- -moz-border-radius-topleft: 10px;
- -moz-border-radius-topright: 10px;

pour obtenir les arrondis sur les coins des bordures.

Remarque : La propriété `border-radius` n'est pas encore supportée par tous les navigateurs. Par conséquent, pour un site Web devant être compatible avec tous les navigateurs, il faudrait utiliser des images de fond pour obtenir le même résultat, ce qui complexifie le travail. De plus, certains validateurs peuvent considérer l'utilisation de cette propriété comme une erreur.

Correction :

```

1 .menu li
2 {
3   display : inline ;
4   list - style - type : none ;
5   border : 1px solid black ;
6   padding : 4px ;
7   background - color : #bbb ;
8   -moz - border - radius - topleft : 10px ;
9   -moz - border - radius - topright : 10px ;
10 }
11
12 .menu a
13 {
14   text - decoration : none ;
15   color : black ;
16 }
17
18 .menu li .active
19 {
20   background - color : white ;
21   border - bottom : 0px ;
22 }
23
24 .menu li :hover
25 {
26   background - color : #ecc ;
27 }

```

Exercice 12 : Utilisation de l'extension Firebug : onglet HTML

Le but de cet exercice est d'apprendre à utiliser l'onglet HTML de l'extension Firebug pour examiner le code XHTML/CSS d'une page Web "élaborée". L'onglet HTML de Firebug permet de visualiser le code XHTML d'une page Web. De plus, sont données dans le cadre de droite toutes les

propriétés CSS s'appliquant à l'élément sélectionné. Il est également possible de modifier le code XHTML ou le code CSS associé. Le navigateur montre alors le rendu obtenu avec ces modifications. Bien évidemment, ces modifications s'appliquent uniquement au niveau du navigateur, les fichiers XHTML et CSS du serveur n'étant pas modifiés. Par ailleurs, grâce à la deuxième icône du menu de Firebug, il est possible de cliquer sur une partie de la page Web pour que le code XHTML associé s'affiche.

Il s'agit maintenant d'utiliser l'extension Firebug pour examiner le code XHTML/CSS de la page Web <http://www.csszengarden.com/tr/francais/> et répondre aux questions suivantes :

1. Quelle balise contient le texte "Le chemin vers l'édification" ? Quelles sont les trois propriétés CSS s'appliquant ? Pour comprendre la propriété qui n'a pas été donnée dans le cours, modifier sa valeur et en déduire son utilité. Donner également tous les ascendants de cette balise. (Le chemin dans l'arbre XHTML de la balise sélectionnée jusqu'à la racine est donné dans le bandeau à côté du bouton "Éditer".)
2. Quelle balise contient le texte "Jardin Zen CSS" en haut à gauche du navigateur ? Quel est son parent ? Modifier le texte "Jardin Zen CSS" dans le code XHTML. Cela modifie-t-il le rendu dans le navigateur ? Pourquoi ? Quel est l'intérêt de coder de cette manière ?
3. De quels éléments est constitué le menu de droite ? Pourquoi est-il à droite ?

Correction :

1. La balise `span`. Les propriétés sont `color`, `font`, `letter-spacing`. (`letter-spacing` indique l'espacement entre les lettres.) Ses ascendants sont `h3`, `div#preamble`, `div#intro`, `div#container`, `body#css-zen-garden` et `html`.
2. La balise `span` enfant de `h1`. Modifier le texte ne change rien car la balise `span` a la propriété `display:none`. L'intérêt est que la page Web reste compréhensible même si l'on applique pas le CSS. En effet, logiquement, "Jardin Zen CSS" est un titre de niveau 1.
3. Le menu de droite est constitué de titres de niveau 3 et de liste non ordonnées. Le menu est à droite car l'élément `div#linkList` contient les propriétés `position:absolute`; et `right:0`; Leur dire de supprimer la propriété `position:absolute`; pour qu'ils voient que le menu passe alors en bas de la page Web et à gauche.

Exercice 13 : Page d'un site Web

Le but de cette exercice est de créer une page Web. Cette page et le code CSS seront par la suite utilisés pour la création d'un site Web dynamique proposant de répondre à un QCM (Questionnaire à Choix Multiple). Créer deux fichiers XHTML et CSS pour obtenir le rendu donné par la figure 2.9. L'image d'en-tête se trouve à l'adresse :

<http://www-lipn.univ-paris13.fr/~lacroix/Documents/TP/IntroWeb/imageEnigme.png>.

Par ailleurs, il est impératif que tous les éléments de la page Web s'affichent, même si l'on n'a pas de CSS associé. Finalement, toute la case de chaque élément du menu doit pouvoir être utilisée pour cliquer sur le lien qu'elle contient.

Créer son propre style CSS tout en gardant le même fichier XHTML afin de modifier l'affichage de la page Web.

Correction :

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
4 <head>
5 <title>SCORES</title>
6 <meta http-equiv="Content-Type"
7       content="text/html; charset=UTF-8" />
8 <link href="style1.css" rel="stylesheet" type="text/css" />
9 </head>
10 <body>
11   <div id="entete">
```

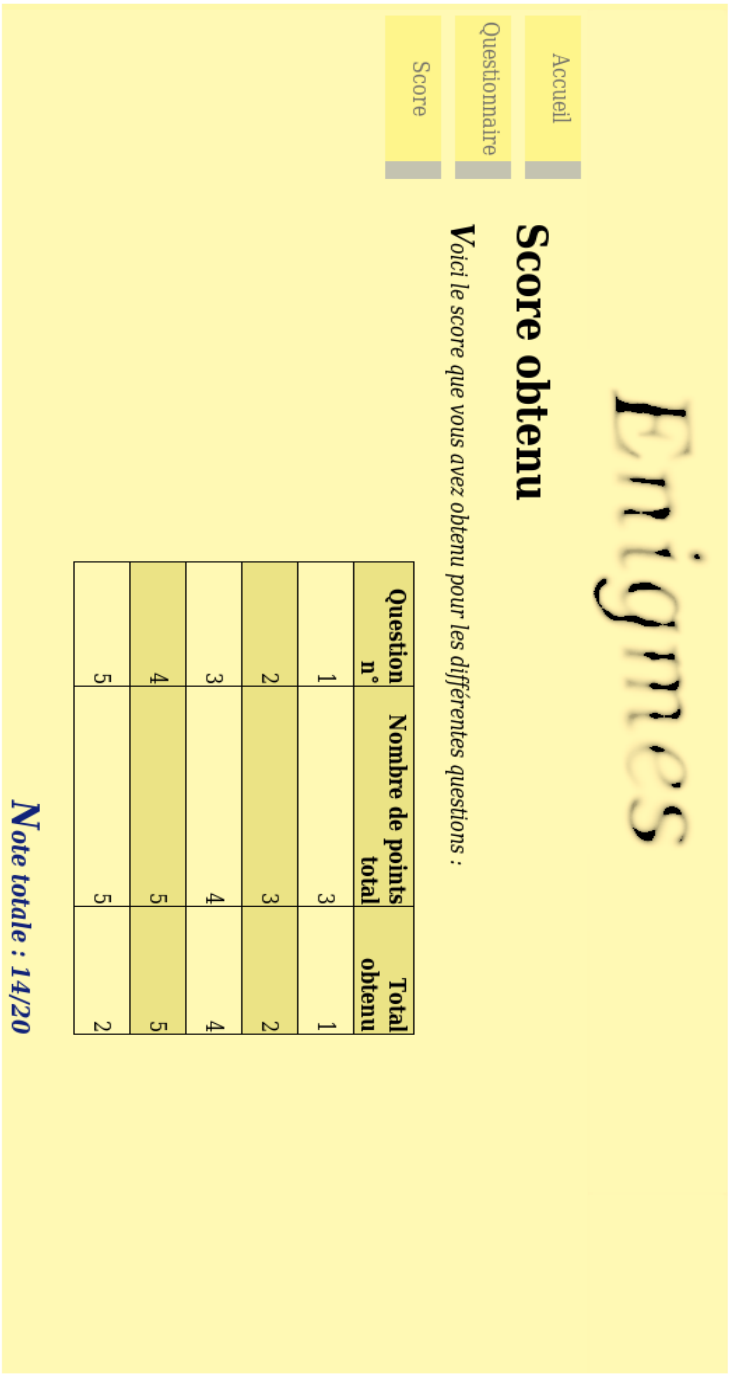


FIGURE 2.9 – Page de score

```
12 <h1> <span> Site d'énigmes </span> </h1>
13 </div>
14
```

```

15     <ul id="menu">
16         <li> <a href="accueil.html"> Accueil </a> </li>
17         <li> <a href="questionnaire.html"> Questionnaire </a> </li>
18         <li> <a href="score.html"> Score </a> </li>
19     </ul>
20
21     <div id="principal">
22         <h1> <span> Score obtenu </span> </h1>
23
24         <p> Voici le score que vous avez obtenu pour les différentes questions : </p>
25         <div class="tableauEtRes">
26 <table>
27     <tr class="impair">
28         <th> Question n </th>
29         <th> Nombre de points total </th>
30         <th> Total obtenu </th>
31     </tr>
32     <tr>
33         <td> 1 </td>
34         <td> 3 </td>
35         <td> 1 </td>
36     </tr>
37     <tr class="impair">
38         <td> 2 </td>
39         <td> 3 </td>
40         <td> 2 </td>
41     </tr>
42     <tr>
43         <td> 3 </td>
44         <td> 4 </td>
45         <td> 4 </td>
46     </tr>
47     <tr class="impair">
48         <td> 4 </td>
49         <td> 5 </td>
50         <td> 5 </td>
51     </tr>
52     <tr>
53         <td> 5 </td>
54         <td> 5 </td>
55         <td> 2 </td>
56     </tr>
57 </table>
58 <p> Note totale : 14/20 </p>
59 </div>
60 </div>
61 </body>
62 </html>

```

```

1 body
2 {
3     border-left : 5px solid #FEF8AA;
4     margin-left : 5%;
5     width : 90%;
6     font-family : "Comic Sans MS";
7     position : relative ;
8     padding-bottom : 100px;
9 }

```



```
10
11 #entete
12 {
13     height:100px; /* comme il n'y a rien, il faut lui donner une taille*/
14     background-image:url(imageEnigme.png);
15     background-color:#FFF9B4;
16     margin : 0;
17     padding : 0;
18 }
19 #entete h1 span
20 {
21     display : none;
22 }
23
24 #entete h1
25 {
26     padding : 0;
27     margin : 0;
28 }
29
30 #principal
31 {
32     background-color:#FFF9B4;
33     padding-left : 180px;
34     padding-top : 20px;
35 }
36
37 #principal h1:first-child
38 {
39     margin : 0;
40 }
41
42 #menu
43 {
44     width:150px;
45     padding : 0px;
46     position : absolute;
47     left : 10px;
48     top : 100px;
49 }
50
51 #menu li
52 {
53     list-style-type :none;
54     text-align : center;
55     margin-bottom : 10px;
56     padding : 0;
57 }
58
59 #menu li a
60 {
61     background-color : #FEF58B;
62     display:block;
63     border-right : 15px solid #C4C3B0;
64     text-decoration : none;
65     height : 30px;
66     padding : 5px;
67     color : #77756B;
```

```

68     margin : 0;
69 }
70
71 table
72 {
73     border-collapse: collapse;
74     margin-left : auto;
75     margin-right : 0px;
76 }
77
78 p
79 {
80     font-style : italic;
81 }
82
83
84 p:first-letter
85 {
86     font-size:1.5em;
87     font-weight:bold;
88 }
89
90 tr
91 {
92     height : 40px;
93 }
94
95 td,th
96 {
97     border : 1px solid black;
98 }
99
100 td
101 {
102     text-align : right;
103 }
104
105 tr.impair
106 {
107     background-color : #EBE385;
108 }
109
110 .tableauEtRes p
111 {
112     font-size : large;
113     font-weight : bold;
114     color : #112279;
115 }
116
117 .tableauEtRes
118 {
119     width : 600px;
120     margin-top : 20px;
121     text-align : right;
122 }

```

◇

Exercice 14 : Page Web pour le cours d'Introduction Web

Le but de cette exercice est de créer une page Web pour le cours d'introduction Web. Créer deux fichiers XHTML et CSS pour obtenir le rendu donné par la figure 2.10. Les documents et

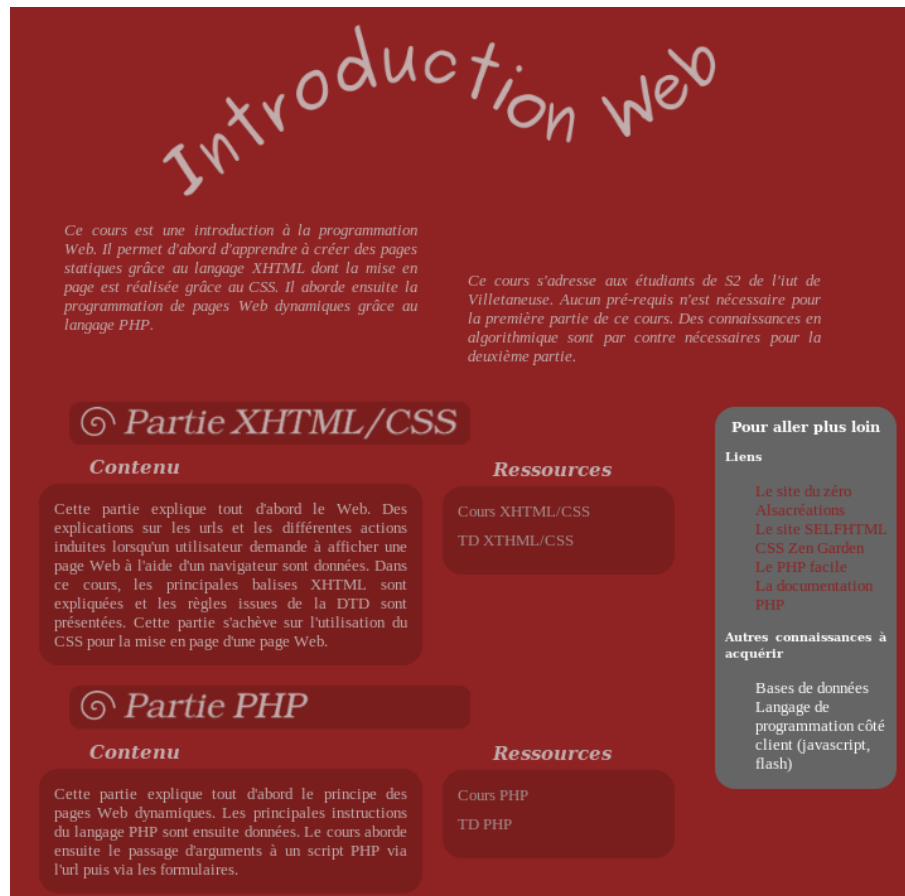


FIGURE 2.10 – Page présentant le cours d'Introduction Web.

images nécessaires à la réalisation se trouvent dans le dossier :

<http://www-lipn.univ-paris13.fr/~lacroix/Documents/IntroWeb/>.

Par ailleurs, il est impératif que tous les éléments de la page Web s'affichent, même si l'on n'a pas de CSS associé. Créer son propre style CSS tout en gardant le même fichier XHTML afin de modifier l'affichage de la page Web.

Correction :

Fichier XHTML :

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr">
4 <head>
5 <meta http-equiv="Content-Type"
6       content="text/html; charset=UTF-8" />
7 <link href="introWeb.css" rel="stylesheet" type="text/css" />
8 <title> Mise en forme de paragraphes et titres</title>

```

```

9 </head>
10 <body>
11
12 <h1> <span> Introduction Web </span></h1>
13
14 <div class="placement">
15 <p class="premierP"> Ce cours est une introduction... </p>
16
17 <p class="deuxiemeP"> Ce cours s'adresse aux étudiants... </p>
18 </div>
19
20 <div class="partie">
21 <h2 id="partieXHTMLCSS"> <span> Partie XHTML/CSS </span> </h2>
22
23 <h3> Contenu </h3>
24
25 <p> Cette partie explique tout d'abord le Web... </p>
26
27 <div class = "aDroite">
28 <h3> Ressources </h3>
29 <ul>
30     <li> <a href="Documents/cours_XHTML_CSS.pdf"> Cours XHTML/CSS </a> </li>
31     <li> <a href="Documents/TP_XHTML_CSS.pdf"> TD XHTML/CSS </a> </li>
32 </ul>
33 </div>
34 </div>
35
36 <div class="partie">
37 <h2 id="partiePHP"> <span> Partie PHP </span> </h2>
38
39 <h3> Contenu </h3>
40
41 <p> Cette partie explique tout d'abord le principe des pages Web... </p>
42 <div class = "aDroite">
43 <h3> Ressources </h3>
44 <ul>
45     <li> <a href="Documents/cours_PHP.pdf"> Cours PHP </a></li>
46     <li> <a href="Documents/TP_PHP.pdf"> TD PHP </a></li>
47 </ul>
48 </div>
49 </div>
50
51 <div id="plusLoin">
52 <h2> Pour aller plus loin </h2>
53
54 <h3> Liens </h3>
55 <ul>
56     <li> <a href="http://www.siteduzero.com/"> Le site du zéro </a> </li>
57     <li> <a href="http://www.alsacreations.com/"> Alsacr  ations </a> </li>
58     <li> <a href="http://fr.selfhtml.org/"> Le site SELFHTML </a> </li>
59     <li> <a href="http://www.csszengarden.com/"> CSS Zen Garden </a> </li>
60     <li> <a href="http://www.lephpfacile.com/cours/"> Le PHP facile </a> </li>
61     <li> <a href="http://www.php.net/docs.php"> La documentation PHP </a> </li>
62 </ul>
63
64 <h3> Autres connaissances    acqu  rir </h3>
65 <ul>
66     <li> Bases de donn  es </li>

```

```

67     <li> Langage de programmation côté client (javascript, flash) </li>
68 </ul>
69 </div>
70 </body>
71 </html>

```

Fichier CSS :

```

1  body
2  {
3      color : #C2ACAC;
4      width : 800px;
5      margin-left : auto;
6      margin-right : auto;
7      font-family : georgia, Verdana, "Times new Roman";
8      background-color : #8F2323;
9  }
10
11  p
12  {
13      text-align : justify ;
14      width : 350px;
15  }
16
17  h1
18  {
19      margin-left : auto;
20      margin-right : auto;
21      background-image : url('Images/titre.png');
22      width : 560px;
23      height : 172px;
24  }
25
26  h1 span, h2 span
27  {
28      display : none;
29  }
30
31  .placement
32  {
33      position : relative ;
34      height : 160px;
35      padding : 0px;
36  }
37
38  .placement p
39  {
40      margin : 0px;
41      padding : 0px;
42      font-style : italic ;
43  }
44
45  .placement .premierP
46  {
47      position : absolute;
48      top : 0px;
49      left : 25px;
50  }
51

```

```
52 .placement .deuxiemeP
53 {
54     position : absolute;
55     top : 50px;
56     right : 25px;
57 }
58
59 .partie
60 {
61     position : relative ;
62 }
63
64 h2#partieXHTMLCSS
65 {
66     background-image : url('Images/partieXHTML_CSS.png');
67     width : 398px;
68     height : 43px;
69     margin-left : 30px;
70 }
71
72 h2#partiePHP
73 {
74     background-image : url('Images/partiePHP.png');
75     width : 398px;
76     height : 43px;
77     margin-left : 30px;
78 }
79
80 .partie h2
81 {
82     margin-bottom : 0px;
83 }
84
85 .partie h3
86 {
87     font-family : Verdana;
88     font-weight : bold;
89     font-style : italic ;
90     padding-left : 50px;
91     margin-bottom : 5px;
92     margin-top : 10px;
93 }
94
95 .partie p, .partie ul
96 {
97     background-color : #791E1D;
98     padding : 15px;
99     -moz-border-radius: 20px;
100     margin-top : 0px;
101 }
102
103 .partie ul
104 {
105     width : 200px;
106     list-style-type : none;
107 }
108
109 .partie ul li
```

```

110 {
111     margin-bottom : 10px;
112 }
113
114 .aDroite
115 {
116     position : absolute;
117     top : 45px;
118     right : 170px;
119 }
120
121 #plusLoin
122 {
123     width : 180px;
124     position : absolute;
125     top : 400px;
126     right : 200px;
127     background-color : #656565;
128     -moz-border-radius: 20px;
129     color : white;
130 }
131
132 #plusLoin h2
133 {
134     text-align : center;
135     font-family : Roman;
136     font-size : 14px;
137 }
138
139 #plusLoin h3
140 {
141     font-family : Verdana;
142     font-size : 12px;
143     padding-left : 10px;
144     text-align : justify ;
145     padding-right : 10px;
146 }
147
148 #plusLoin li
149 {
150     list-style-type : none;
151     font-size : 16px;
152 }
153
154 #plusLoin a
155 {
156     text-decoration : none;
157     color : #8F2323;
158 }
159
160 #plusLoin a:hover
161 {
162     font-style : italic ;
163     color : #BC1919;
164 }
165
166 .partie ul li a
167 {

```

```
168         text-decoration : none;
169         color : #A59090;
170     }
171
172     .partie ul li a:hover
173     {
174         font-style : italic ;
175         color : #774343;
176     }
```