



PREMIERE GENERALE

NUMERIQUE ET SCIENCES INFORMATIQUES

LANGAGE HTML

▷ Histoire de l'informatique

- ▷ Représentation des données : types et valeurs de base
- ▷ Représentation des données : types construits
- ▷ Traitement de données en tables

▷ Interactions entre l'homme et la machine sur le Web

- ▷ Architectures matérielles et systèmes d'exploitation
- ▷ Langages et programmation
- ▷ Algorithmique



Tim Berners-Lee, informaticien du CERN, est à l'origine au début des années 1990, du « World Wide Web », plus communément appelé « Web » : en compagnie de l'ingénieur Robert Cailliau.

Il s'agit au départ d'un projet permettant l'échange et le partage de l'information avec ses collaborateurs. Il évoque l'idée d'une « toile » où chacun pourrait « naviguer » de contenus en contenus. Le principe est de pouvoir relier les postes informatiques des scientifiques, via les réseaux informatiques, en un système d'information puissant, mais surtout simple à faire fonctionner. En 1990, pour faire fonctionner le système, Tim Berners-Lee met au point le **protocole HTTP** (Hyper Text Transfer Protocol), ainsi que le **langage HTML** (HyperText Markup Language).

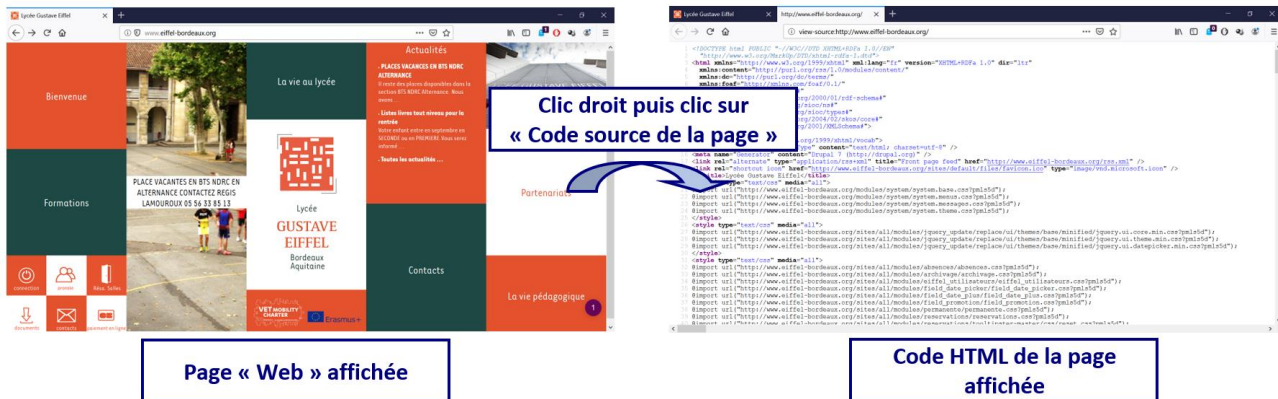
1 – MISE EN SITUATION

1.1 – Page Web

« Une **page Web** est une ressource du World Wide Web conçue pour être consultée par des visiteurs à l'aide d'un **navigateur Web** (Internet Explorer, Mozilla Firefox etc.). Elle a une adresse Web. Techniquement, une page Web est souvent constituée d'un document en **Hypertext Mark-Up Language (HTML)** et d'images. Cependant, tout type de ressources ou d'assemblage de ressources, textuelles, visuelles, sonores, logicielles, peuvent constituer une page Web. » (Wikipédia)

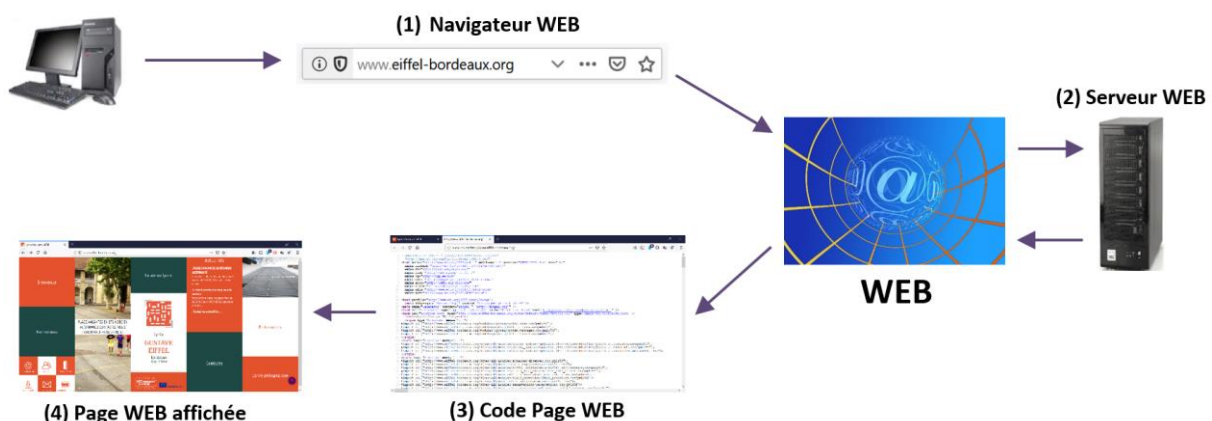
1.2 – HyperText Markup Language (HTML)

Le langage **HTML** est un langage de description permettant de **représenter les pages web**. Il permet de **structurer sémantiquement** et de **mettre en forme** le contenu des pages, d'**inclure** des ressources multimédias dont des **images**, des **vidéos**, des **applets**...



1.3 – Protocole http (communication client/serveur)

Un serveur web est un ordinateur distant permettant à des utilisateurs (clients) d'effectuer des requêtes et afficher des pages internet. Il met en œuvre le protocole applicatif HTTP. D'un point de vue utilisateur, cela nécessite un navigateur à partir duquel on **envoie une requête (1) à travers une adresse WEB (URL)**. Cette adresse est en réalité l'adresse IP du serveur distant, un protocole (le protocole DNS) permet le mécanisme d'association de cette adresse IP à l'adresse WEB. Le **serveur WEB reçoit cette requête à travers WEB (2)**. Si la page demandée existe, le **serveur répond en renvoyant au navigateur le code html de la page demandée (3)**. Le navigateur **interprète le code et affiche la page Web (4)**.



2 – LANGUAGE (HTML)

Le langage HTML un **langage de balisage**, c'est-à-dire que le texte (le contenu) est **structuré par des balises** qui en **définissent la structure** (le contenant).

Il s'agit d'un **langage hypertexte**. Cela signifie qu'il est possible de définir **des liens entre plusieurs documents** ou **au sein d'un même document**. Les documents pointés par les liens peuvent se situer sur des machines éloignées, faisant partie d'Internet.

Les pages web sont généralement **organisées autour d'une page d'accueil**, jouant un point central dans la navigation à l'aide des liens hypertextes. Cet ensemble cohérent de pages web liées par des liens hypertextes et articulées autour d'une page d'accueil commune est appelée **site web**.

Le langage HTML permet de créer des documents pouvant **être lus par des équipements très variés** conformément aux **exigences de l'accessibilité du web**. Il est souvent utilisé conjointement avec des **langages de programmation (JavaScript)** et des **formats de présentation (feuilles de style ou CSS)**.

2.1 – Balises (ou conteneurs)

La description d'un document HTML passe par l'utilisation de **balises** (ou **tags** en anglais). La plupart des balises modifient l'interprétation que fait le navigateur sur leur contenu. Puisque les balises **délimitent du contenu** elles sont faites d'une **partie ouvrante** et d'une **partie fermante**.

Les balises ouvrantes sont faites d'une instruction entre < et >. Les balises fermantes sont faites de la même instruction, mais entre </ et >. Par exemple pour définir un paragraphe on utilise la **balise** <p> :

Balise ouvrante <p> : début du paragraphe **Balise fermante </p> : fin du paragraphe**

<p> Paragraphe </p>

Une liste non exhaustive des balises HTML est disponible sur les sites suivants :

<http://www.w3schools.com/tags/default.asp> ou <http://www.w3.org/TR/REC-html40/index/elements.html>

2.2 – Marqueurs

Toutes les balises ne sont pas destinées à contenir du texte. Certaines ne sont que de simples marqueurs (saut de ligne, insertion d'une image...). Ce sont des **balises orphelines**. Par exemple le saut de ligne peut être écrit simplement **
** (abréviation de break).

2.3 – Attributs

Les **attributs servent à modifier les propriétés des balises**. Ils sont optionnels. Les attributs se placent après l'instruction de la balise. Ils sont suivis du signe égal =, puis d'une valeur entre doubles guillemets. Par exemple, la ligne de code suivante permet coloriser le fond d'écran (bgcolor) en noir.

```
<body bgcolor = "black">
```

3 – STRUCTURE D'UNE PAGE HTML

Toutes les pages HTML utilisent la même structure de base, dont voici les ingrédients:

```
<!-- commentaire -->
<html>
  <head>

  .....

</head>
<body>

.....

</body>
</html>
```



3.1 – Commentaires

Les commentaires peuvent être placés n'importe où dans un document HTML à condition de ne pas être imbriqués. Ils sont placés entre les chaînes de caractère `<!-- et -->`.

```
<!-- Ceci est un commentaire -->
```

3.2 – Balise `<html>`

C'est le premier marqueur ou tag que l'on trouve dans un document HTML. La totalité du document est contenue entre les balises `<html>` et `</html>`.

3.3 – Balise `<head>`

L'en-tête du document est réservée aux méta-informations (les informations relatives au document) c'est-à-dire des informations n'apparaissant pas sur la page HTML, mais la décrivant. Ces informations sont utilisées par exemple par les moteurs de recherche pour indexer les pages Web. Ce conteneur inclut en particulier la balise de titre `<title>`. Ce titre n'est pas celui qui apparaît sur la page, mais celui qui est utilisé par les moteurs de recherche, et dans la barre de navigation.

3.4 – Balise `<body>`

Le conteneur `<body>` contient ce qui doit être affiché: texte, images, sons...

Bien que l'utilisation des feuilles de style rende obsolète cette technique, il est possible d'affecter des attributs à la balise `<body>`. La balise body accepte plusieurs attributs :

- **text** : permet de changer la **couleur du texte** de l'ensemble de la page WEB ;
- **bgcolor** : permet de changer la couleur du fond de la page ;
- **background** : permet de spécifier une **image de fond**.

Les couleurs peuvent être spécifiées de deux façons :

- par un **nom prédéfini** : red, green, blue, yellow, white, black... ;
- par un **triplet RGB** : un triplet est spécifié par un dièse (#) suivi de trois nombres à deux chiffres. Les chiffres sont en hexadécimal (0, 1, ... 9, A, B, ... , F) représentant respectivement la composante rouge, vert et bleu.

Exemple : couleur du texte en rouge

```
<body text="red"> ou <body text="#FF0000">
```

Exemple : image en fond d'écran

```
<body background="fond_ecran.jpg">
```

Exercice 1 : Structure d'une page HTML

1. **Donner** la balise permettant de délimiter l'ensemble du code HTML.
2. **Citer** les deux grands blocs de code HTML qui constituent un fichier HTML.
3. **Donner** pour chacun de ces deux blocs, le rôle et la balise permettant de le délimiter.

Pour **visualiser la page WEB**, il est nécessaire d'ouvrir le fichier HTML à l'aide d'un **navigateur WEB** (ou browser) tel que Firefox, Internet Explorer, Chrome, Safari... Le rendu peut ne pas être identique selon le navigateur choisi.



Pour **visualiser et modifier le code HTML**, il est nécessaire d'ouvrir le fichier HTML à l'aide d'un **éditeur de texte** tel que Notepad++, Geany, Fraise (MAC) ... Cependant un éditeur de texte simple tel que Bloc Note peut suffire. Nous utiliserons le logiciel **Visual Studio Code**.

Exercice 2 : Balises utilisées dans l'entête

1. **Ouvrir**, avec **Firefox**, le fichier « index.html » qui se trouve dans le dossier « Fichiers TP HTML ».
2. **Ouvrir** le fichier « index.html » à l'aide du logiciel **Visual Studio Code**.
3. **Supprimer** la quatrième ligne `<title>NUMERIQUE ET SCIENCES INFORMATIQUES</title>`.
4. **Actualiser**, sous FireFox, l'affichage de la page WEB en appuyant sur la touche « F5 » du clavier.
5. **Justifier**, à partir de la modification de la page WEB, le rôle de la balise `<title>`.
6. **Remplacer** la ligne `<title>NUMERIQUE ET SCIENCES INFORMATIQUES</title>`. **Modifier-la** pour faire apparaître l'information suivante : **NSI au Lycée Gustave Eiffel**.
7. **Vérifier** que la modification est effective.

4 – FORMATAGE DU TEXTE

4.1 – Formatage du texte

La majeure partie d'une page HTML est constituée de texte. Cependant, le navigateur n'interprète pas le texte comme le ferait un éditeur :

- les tabulations sont transformées en espaces ;
- plusieurs espaces successifs sont regroupés en un seul ;
- les sauts de lignes sont ignorés.

4.2 – Titres, paragraphes, etc.

Les titres sont insérés des balises de titre `<h1>` à `<h6>`. La balise `<h1>` définit le **titre de plus haut niveau**.

Créer un paragraphe revient à placer du texte dans une balise `<p>`.

Pour forcer un **retour à la ligne** on utilise `<br>`.

Enfin, pour placer une **ligne séparatrice horizontale** entre deux parties de la page on utilise `<hr>` (Horizontal Rule).

Les balises `<p>` et `<h1>` à `<h6>` admettent l'**attribut align** qui permet de définir le **type d'alignement horizontal** du texte. Il peut prendre les valeurs `left` (par défaut), `right`, `center` ou `justify`.

Exemple : centrer un titre

```
<h1 align="center"> NUMERIQUE ET SCIENCES INFORMATIQUES </h1>
```

4.3 – Gras, italique, etc. (obsolète ou sémantique changée en HTML5)

HTML fournit plusieurs balises très utilisées pour faire ressortir du texte. Ces balises sont cependant obsolètes en HTML5 : `<b>` : texte en **gras** ; `<i>` : texte en **italique** ; `<em>` : **souligner** le texte ; `<sup>` et `<sub>` : du texte en **exposant** ou en **indice**.



Exemple : Mettre en gras du texte

```
<p><b>L'horaire de cet enseignement de spécialité est de 3 heures par semaine</b> en salle informatique (un ordinateur pour chaque élève).</p>
```

L'horaire de cet enseignement de spécialité est de 3 heures par semaine
en salle informatique (un ordinateur pour chaque élève).

4.4 – Balise (obsolète en HTML5)

La balise permet de modifier la taille (attribut **size**), la police (attribut **face**) et la couleur du texte (attribut **color**). Ces balises sont cependant obsolètes en HTML5.

La valeur de l'attribut **size** est comprise entre 1 et 7, la valeur 3 étant la taille par défaut. La taille réelle des polices dépend toutefois des options et de la résolution choisie par chaque utilisateur.

Exemple : Mise en forme du titre <h1>

```
<h1><font size="6" face="lucida handwriting" color="red"> Numérique </font> et Sciences Informatiques </h1>
```

Numérique et Sciences Informatiques

4.5 – Lister, énumérer...

Le langage HTML permet d'arranger des éléments en listes ordonnées ou en items. Une **liste non ordonnée** est créée avec la balise (unordered list) et une **liste ordonnée** avec (ordered list). Chaque élément d'une liste est délimité par la balise (list item).

Il est possible de modifier l'apparence de la liste non ordonnée avec l'attribut **type** dont la valeur peut être **disc**, **circle** ou **square**.

Exemple : Liste non ordonnée et ordonnée

```
<ul type="square">  
  <li>item</li>  
  <li>item</li>  
</ul>
```

- item
- item

```
<br>  
<ol>  
  <li>élément 1</li>  
  <li>élément 2</li>  
</ol>
```

1. élément 1
2. élément 2

Exercice 3 : Formatage du texte

1. **Remplacer**, à la ligne 16 et 20, les balises <p align="justify"> et </p> par <h1 align="justify"> et </h1>.
2. **Donner** le rôle des balises <p> et <h1> à <h6>.
3. **Annuler** les modifications précédentes.
4. **Supprimer** les balises
 à la fin des lignes 9 et 10.
5. **Justifier**, à partir de la modification de la page WEB, le rôle de la balise
.
6. **Annuler** les modifications précédentes.

Exercice 3 : Formatage du texte

7. **Supprimer** la ligne 13 `<hr size="2" color="#0000127">`.
8. **Justifier**, à partir de la modification de la page WEB, le rôle de la balise `<hr>`.
9. **Annuler** la modification précédente.
10. **Modifier** à la ligne 8, les attributs de la balise `<font>` afin que la police de caractère utilisée soit la police "lucida handwriting" et la couleur soit une couleur verte.
11. **Justifier**, à partir de la modification de la page WEB, le rôle de la balise `<font>`.
12. **Annuler** la modification précédente.
13. **Remplacer**, à la ligne 35 et 44, les balises `<ul>` et `</ul>` par `<ol>` et `</ol>`.
14. **Justifier**, à partir de la modification de la page WEB, le rôle des balises `<ul>` et `<ol>`.
15. **Annuler** les modifications précédentes.

Les **attributs servent à modifier les propriétés des balises**. Ils sont optionnels. Les attributs se placent après l'instruction de la balise. Ils sont suivis du signe égal =, puis d'une valeur entre doubles guillemets. Par exemple, `<body bgcolor="black">` permet coloriser le fond d'écran en noir. Avec HTML5, utilisation de la grande majorité des attributs n'est plus conseillé.

Exercice 4 : Attributs

1. **Remplacer**, à la ligne 9, `<h1 align="center">` `<font size="7">` Numérique et Sciences Informatiques `<br>`, pour l'attribut `align` la valeur "center" par "right".
2. **Justifier**, à partir de la modification de la page WEB, le rôle de l'attribut `align`.
3. **Annuler** la modification précédente.
4. **Modifier** le fichier HTML afin que le texte des paragraphes soit justifié.
5. **Modifier** les attributs de la balise `<font>` afin que l'ensemble du texte de la page WEB soit en police arial et de couleur rouge.
6. **Annuler** les modifications précédentes.
7. **Ajouter** un attribut à la balise `<ul>` afin que les puces devant chaque item de la liste soit un carré.

5 – INSERTION D'IMAGES DANS LE TEXTE

Des images peuvent être insérées dans le texte d'un document HTML. Elles peuvent être au format **GIF**, **BMP**, **PNG** ou **JPG**. Elles peuvent également servir pour des liens hypertextes :

`` pour **insérer une image locale**, où `nom_image.gif` est le nom complet du fichier avec éventuellement son chemin relatif à partir du répertoire du document HTML.

`` pour insérer une **image distante**. URL étant l'adresse complète du fichier image.



Exercice 5 : Images

1. **Modifier**, dans la ligne 11 `` les valeurs des attributs `width` et `height`.
2. **Justifier**, à partir de la modification de la page WEB, le rôle de ces attributs.
3. **Remplacer** l'image du lycée EIFFEL par l'image « Lycee_Eiffel_2.jpg » qui se trouve dans le dossier Images.
4. **Rajouter** des images pour illustrer la page WEB.

6 – LIENS HYPERTEXTES

Le langage HTML est un langage hypertexte. Cela signifie qu'il est possible de rompre la linéarité du texte pour se **diriger vers une autre page web**, ou **tout autre type de ressource informatique**. Ceci est effectué grâce à un **lien** : c'est à dire la mise en relation entre deux parties du World Wide Web, une source et une destination. Un lien peut pointer sur tout type d'objet informatique, pas uniquement sur une page web. En HTML, un lien est introduit grâce à la **balise** `<a>`.

6.1 – Les liens à l'intérieur de la page elle-même

Pour créer la destination d'un lien à l'intérieur d'une page on utilise l'attribut `name` de la balise `<a>`. Par exemple `<a name="truc">`. Il est ensuite possible d'accéder à ce lien à partir d'un autre point de la page en créant un lien source ainsi : `<a href="#truc">`.

6.2 – Les liens sur des pages ou ressources d'un même site

Il est aussi possible de créer un **lien sur une autre page dans le même répertoire** que la page web en cours. Par exemple si le répertoire courant contient deux pages PageA.html et PageB.html. La page A peut pointer sur la page B si elle contient un lien de la forme :

```
<a href="PageB.html">La page B</a>
```

6.3 – Les liens sur des pages ou ressources d'autres sites

Pour diriger un lien vers une page arbitraire du World Wide Web, il est nécessaire, en plus du nom de la page d'ajouter le protocole, et le nom du serveur Web contenant la page. Un tel chemin d'accès est nommé URL.

Voici un lien qui amène vers le site du lycée Gustave Eiffel :

```
<a href="http://www.eiffel-bordeaux.org/">Lycée Gustave Eiffel</a>
```

6.4 – Couleurs des liens

Il est possible de définir la couleur des liens, des liens actifs et des liens visités dans le fichier HTML. Cependant pour la version HTML5, il est conseillé de configurer les liens dans la feuille de style. Le lien devient **actif** lorsque l'on clique dessus. Le lien est **visité** lorsque la page spécifiée par le lien a été ouverte (et qu'elle se trouve encore dans la mémoire cache du navigateur).

Pour définir les couleurs d'un lien il faut placer les attributs `link` (couleur du lien), `aLink` (couleur du lien actif) et `vLink` (couleur du lien visité) dans la balise `body`.



Exemple :

```
<body link="#00007F" alink="red" vlink="#00FF00">
```

```
<a href="http://www.eiffel-bordeaux.org/">Lycée Gustave Eiffel</a>
```

Lien : [Lycée Gustave Eiffel](#)

Lien actif : [Lycée Gustave Eiffel](#)

Lien visité : [Lycée Gustave Eiffel](#)

6.5 – Ouverture du lien dans une nouvelle page

Par défaut ; lorsque l'on clique sur un lien, la page cible s'ouvre en lieu et place de la page courante. Pour **ouvrir la page cible dans une nouvelle fenêtre**, on utilise l'attribut **target** avec la valeur **_blank**, dans la balise **<a>**.

```
<a href="Page2.html" target="_blank"> Page suivante </a>
```

6.6 – Infobulles

L'attribut **title** de la balise **<a>** permet de faire afficher un court descriptif du lien lorsque le curseur de la souris passe sur le lien.

```
<a href="http://www.eiffel-bordeaux.org/" title="Site du lycée Gustave Eiffel"> Lycée  
Gustave Eiffel </a>
```

[Lycée Gustave Eiffel](#)

Site du lycée Gustave Eiffel

Exercice 6 : Liens hypertextes vers la même page

1. **Donner** la balise permettant de réaliser un lien hypertexte.
2. **Ouvrir**, avec **Firefox**, le fichier « index.html ».
3. **Cliquer** sur le lien « Bas de la page ». **Décrire** l'action réalisée.
4. **Ouvrir** le fichier « accueil.html » à l'aide du logiciel **Visual Studio Code**.
5. **Noter** les 2 lignes de codes permettant de réaliser ce lien hypertexte.
6. **Réaliser** de la même manière un lien hypertexte permettant de revenir en haut de la page.
7. **Modifier** les couleurs des liens afin qu'ils aient la même couleur que le texte et une couleur différente lorsqu'ils ont été cliqués.
8. **Remplacer** le texte des liens par des images (flèche descendante et montante par exemple). **Rajouter** une description du lien lorsque la souris passe dessus.

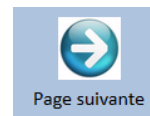
Numérique et Sciences Informatiques
Lycée Gustave Eiffel



[Bas de la page](#)

Exercice 7 : Liens hypertextes vers une autre page

1. **Cliquer** sur la flèche en bas de la page. **Décrire** l'action réalisée.
2. **Ouvrir** le fichier « Page2.html » à l'aide du logiciel **Visual Studio Code**.
3. **Placer** un lien hypertexte permettant de revenir à la page « Index ».
4. **Ajouter** une description du lien lorsque la souris passe dessus ce lien.



Exercice 8 : Liens hypertextes vers des pages sur d'autres sites WEB

La « Page2.html » contient des liens vers d'autres sites WEB dans la section « Liens » en bas de la page.

1. **Modifier** les liens afin que lorsque l'on clique sur le lien, la page s'ouvre dans une nouvelle fenêtre.
2. **Ajouter** une description du lien lorsque la souris passe dessus ces liens.
3. **Remplacer** le lien de la page du Lycée Eiffel par un lien vers le blog ISN du Lycée Eiffel <https://lewebpedagogique.com/isneiffel/>.

7 – TABLEAUX

En HTML, un tableau est défini comme le regroupement d'un ensemble de **lignes**. Chaque ligne est composée d'un ensemble de cases appelées **cellules**. Le début d'un tableau est défini par la balise `<table>`, alors que la balise `</table>` le termine.

La balise `<tr>` introduit une nouvelle ligne, alors que la balise `</tr>` termine cette ligne. Une cellule est définie par les balises `<td></td>`.

```
<table>
  <tr>                <!-- début de la première ligne -->
    <td> Nom </td>      <!-- première cellule-->
    <td>Prénom </td>    <!-- deuxième cellule-->
  </tr>                <!-- fin de la première ligne -->
  <tr>                <!-- début de la deuxième ligne -->
    <td>Rémi</td>       <!-- première cellule-->
    <td>Georges</td>    <!-- deuxième cellule-->
  </tr>                <!-- fin de la deuxième ligne -->
</table>
```

Nom	Prénom
Rémi	Georges

Pour **encadrer les cellules** du tableau, appliquez l'attribut `border` à la balise `<table>`.

```
<table border='1'>
```

Nom	Prénom
Rémi	Georges

Il est possible d'**aligner horizontalement** le texte en appliquant les valeurs `left`, `right` et `center` à l'attribut `align`. Pour **aligner verticalement**, il faut utiliser l'attribut `valign` avec les valeurs `top`, `middle` et `bottom`.

Il est possible de changer la couleur de chaque cellule avec l'attribut `bgcolor`.



Les lignes des tableaux réalisés précédemment avaient le même nombre de cellules. Il est cependant très fréquent de devoir **définir une cellule sur plusieurs lignes ou sur plusieurs colonnes**. Pour cela il faut fusionner des cellules. On utilise les attributs **colspan** (on étend une cellule sur plusieurs colonnes) et **rowspan** (on étend une cellule sur plusieurs lignes).

```
<table valign='center' border='1'>
  <tr>
    <td colspan='2'>Titre</td>      <!-- cellule sur 2 colonnes>
  </tr>
  <tr>
    <td>Paragraphe 1 </td>
    <td rowspan='2'>Image</td>      <!-- Cellule sur 2 lignes>
  </tr>
  <tr>
    <td bgcolor='red'>Paragraphe 2</td> <!-- Cellule coloriée>
  </tr>
</table>
```

Titre	
Paragraphe 1	Image
Paragraphe 2	

Exercice 9 : Tableaux

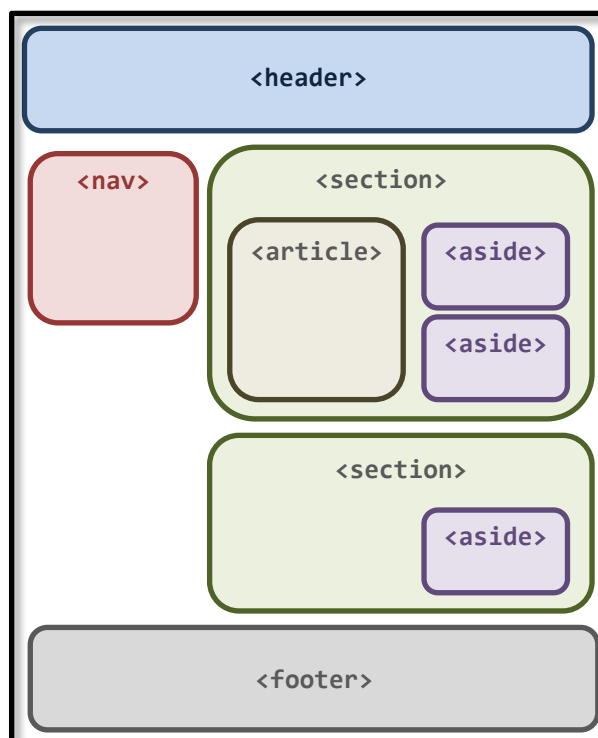
La rubrique « Liens » est organisée sous la forme d'un tableau. On désire ajouter deux autres liens :

- Le site w3schools : <http://www.w3schools.com/tags/default.asp>
- Un lien vers le livre numérique « Apprendre à programmer avec Python 3 » : <http://inforef.be/swi/python.htm>

Modifier le tableau précédent afin d'intégrer ces deux liens.

8 – STRUCTURER UNE PAGE WEB

HTML5 a permis l'introduction de nouvelles balises HTML dédiées à la structuration du site. Les principales balises de structuration sont :



8.1 – L'entête : <header>

La balise <header> permet de définir un entête dans lequel on trouve le plus souvent, un logo, une bannière, le slogan ou le titre du site... On peut placer dans cet entête des balises de types <h>, <p>, ...

```
<header align="center"> <h2>Numérique et Sciences Informatiques</h2>
  </font>
</header>
```



8.2 – Le pied de page : <footer>

Contrairement à l'entête le pied de page se trouve généralement en bas du document. On y trouve généralement des informations telles que les liens de contact, le nom de l'auteur, les mentions légales...

```
<footer align="center">
  <table>
    <tr>
      <td><h4>Contacts et Accès</h4>
      <p>Lycée Gustave Eiffel<br>
        143 cours de la Marne<br>
        33000 Bordeaux</p>
      </td>
      <td><h4>Suivez nous sur Twitter</h4>
      <a href = "https://twitter.com/eiffel_lycee"></a>
      </td>
    </tr>
  </table>
</footer>
```



8.3 – Principaux liens de navigation : <nav>

La balise <nav> permet de contenir les principaux liens de navigation du site (menu principal du site).

```
<nav>
  <ul>
    <li><a href="index.html">Accueil</a></li>
    <li><a href="forum.html">Forum</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</nav>
```

- [Accueil](#)
- [Forum](#)
- [Contact](#)

8.4 – Section de page : <section>

La balise <section> sert à regrouper des contenus en fonction de leur thématique. Chaque section peut avoir son titre de niveau 1 (<h1>). Chacun de ces blocs étant indépendant des autres, il est possible de retrouver plusieurs titres <h1> dans le code de la page web.

```
<section>
  <h2> Un enseignement de l'informatique au lycée </h2>
  <p align="justify">  Les sciences informatiques,
  et plus généralement, les sciences du numérique, ont aujourd'hui envahi nos vies
  professionnelles et personnelles. Elles ont entraîné des mutations profondes dans
  nos sociétés (culture, sciences, économie, politique...)</p>
</section>
```

Un enseignement de l'informatique au lycée



Les sciences informatiques, et plus généralement, les sciences du numérique, ont aujourd'hui envahi nos vies professionnelles et personnelles. Elles ont entraîné des mutations profondes dans nos sociétés (culture, sciences, économie, politique...). Les lycéens sont, aujourd'hui, des usagers intensifs des outils informatiques mais très peu d'entre eux connaissent les mécanismes fondamentaux de l'informatique. L'objectif de la spécialité NSI est l'appropriation des concepts et des méthodes qui fondent l'informatique.

8.5 – Informations complémentaires <aside>

La balise <aside> est conçue pour contenir des informations complémentaires au document que l'on visualise. Ces informations sont généralement placées sur le côté (bien que ce ne soit pas une obligation). Il peut y avoir plusieurs blocs <aside> dans la page.

8.6 – Article indépendant : <article>

La balise <article> sert à englober une portion généralement autonome de la page. C'est une partie de la page qui pourrait ainsi être reprise sur un autre site. C'est le cas par exemple des actualités (articles de journaux ou de blogs).

Exercice 10 : Amélioration de la page web

Modifier la page « Page2.html » afin de l'organiser comme un tableau ayant la forme suivante :

Titre de la page (avec l'image du lycée Gustave Eiffel)	
Corps de la page (différents paragraphes)	Rubrique « liens »
Lien de retour vers la page d'accueil	