



HTML 5.0

JavaScript

PHP (JSP)

MySQL



HTML

5.0

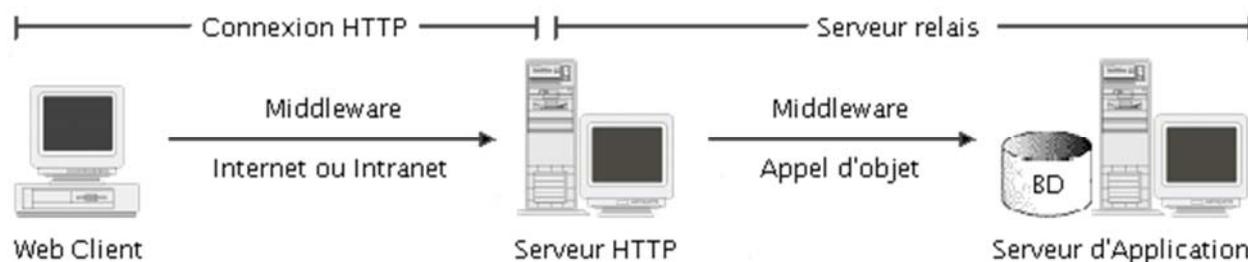


Les standards du Web

- **HTTP** (*HyperText Transfer Protocol*) est un protocole de communication et d'échange de données Client-Serveur.
- **HTML** (*HyperText Markup Language*) norme générique pour l'échange de données multimédias.
 - Un document HTML se compose d'un ensemble de balises délimitant des blocs de texte, en leur conférant des propriétés spécifiques à la visualisation (présentation) par un navigateur universel (IE, Firefox, Chrome, Opera, Safari).
 - Handicap majeur: la structuration sémantique du document n'est pas prise en compte.
 - Dernière norme standardisée HTML 5.0 essaye de palier à ce problème
- **CGI** (*Common Gateway Interface*) est la norme qui définit l'interface permettant l'exécution de programmes externes par un serveur HTTP.
 - Réalisation en mode ISAPI sur les serveurs récents.

Le Client - Serveur Universel

- Le client gère l'interface et contrôle la saisie.
- Le serveur HTTP relaie les données entre le client et le serveur.
- Le serveur d'application se charge du traitement des données.
- Le serveur de base de données se charge d'accéder aux données.



Le modèle Client-Serveur - *thin client*

- "**Client léger**" désigne une application accessible via une interface Web consultable à l'aide d'un navigateur Web (universel):
 - Firefox, IE, Opera, Safari, Google Chrome
- Un "**client léger**" traite des pages Web simple:
 - HTML, CSS, Javascript ou VBScript
- L'essentiel des traitements sont réalisés du côté du serveur et l'interface graphique est envoyée au navigateur à chaque requête (demande du client).
 - *Point négatif*: Surcharge du serveur
 - *Point positif*: Technique AJAX permettant des interventions dynamiques sur le client sans recharge de page.

Le modèle Client-Serveur - *heavy client*

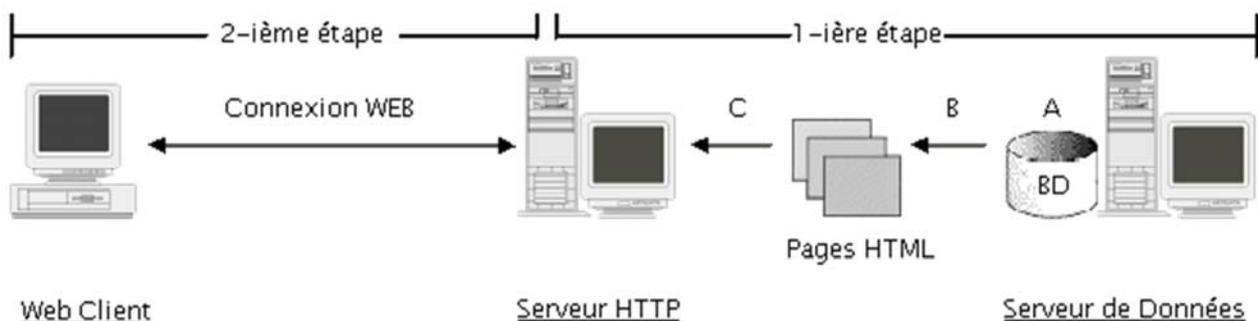
- Le "**Client lourd**" est une application graphique exécutée sur le système d'exploitation de l'utilisateur (côté client).
- Application généralement installée sous le système d'exploitation de l'utilisateur.
- La gestion des mises à jour se fait par Internet afin de la faire évoluer.
- Excellentes fonctionnalités et rapidité dans un environnement intégré d'un système d'exploitation.
- Maintenances et mises à jour demandent une connexion réseau permanente.
- Existe des solutions de gestion à distances (problème de sécurité)
- *Point négatif*: Souvent formats de données propriétaires
- *Point négatif*: Exige la formation des clients utilisateurs

Le modèle Client-Serveur - *mid-heavy client*

- Un "**Client semi-lourd**" est une **applet Java** qui évolue dans le cadre d'un navigateur et est gérée par la machine virtuelle de ce dernier.
- Interface graphique bien développée;
- Problèmes de sécurité résolus;
- Rapidité satisfaisante, premier chargement long;
- Maintenance et mises à jour faciles à distances;
- Le langage Java autorise ce genre d'application.
- Le langage Python évolue dans cette direction
- Dans cette catégorie il faut mentionner aussi les composants téléchargeables ActiveX (Microsoft) et Flash (Adobe).
 - Très rependus sur le Web mais ils ne sont pas standardisés par W3C

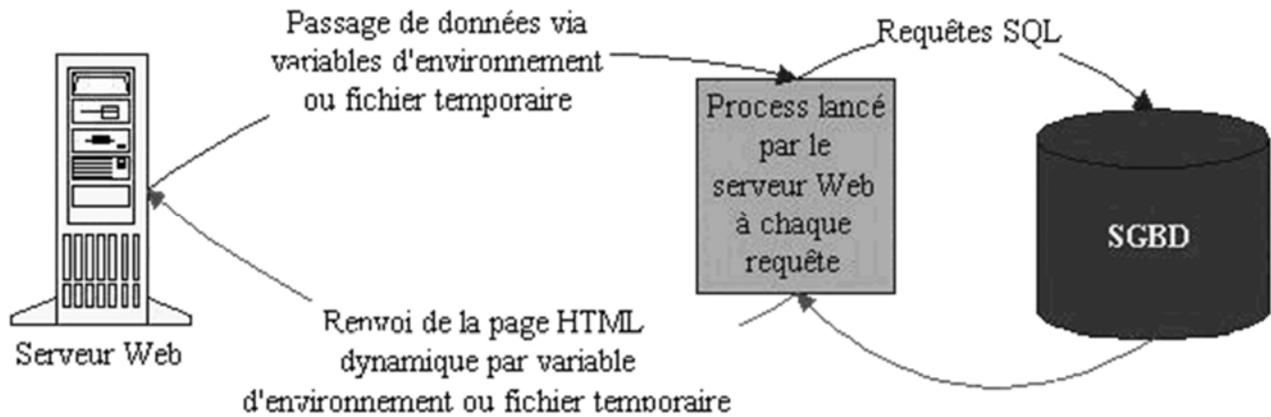
Le modèle Client-Serveur

- **La gestion des pages statiques**
 - Des pages HTML sont générées de manière statiques depuis une base de données avec une certaine régularité (hebdomadaire, mensuel, ...).
 - Une procédure exporte un contenu d'une BD vers l'espace public du serveur Web



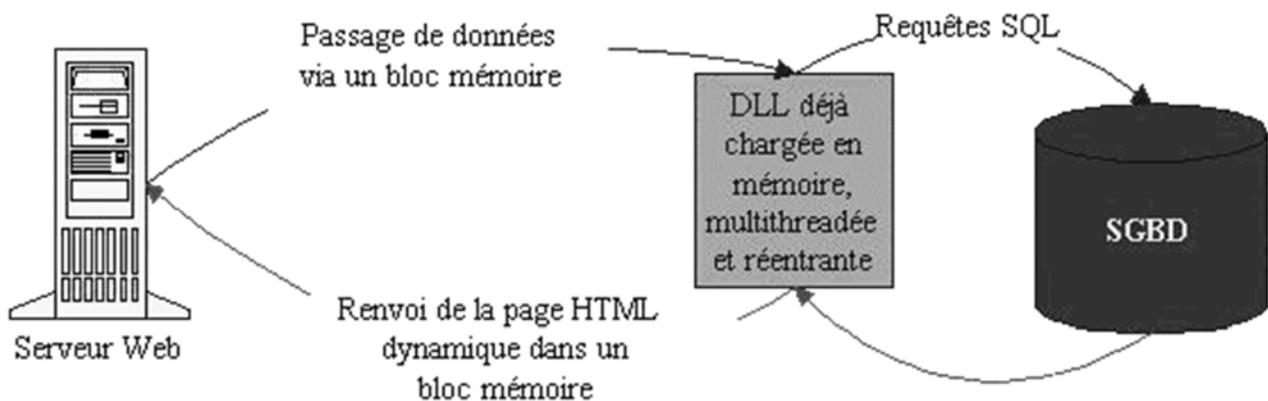
Le modèle Client-Serveur

- **La gestion des pages dynamiques:**
 - Solution à la base de **CGI**:



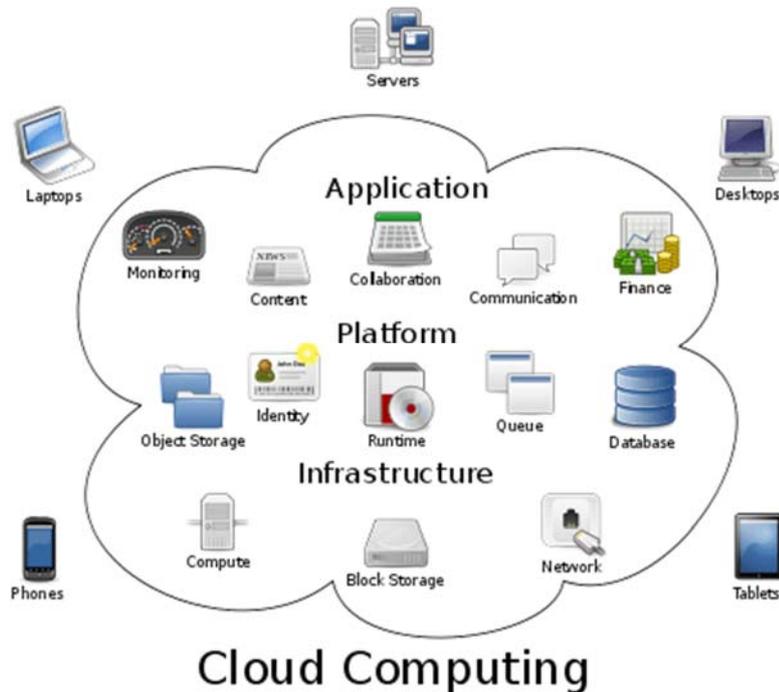
Le modèle Client-Serveur

- **La gestion des pages dynamiques:**
 - Solution à la base de **API** (le modèle **ISAPI**):



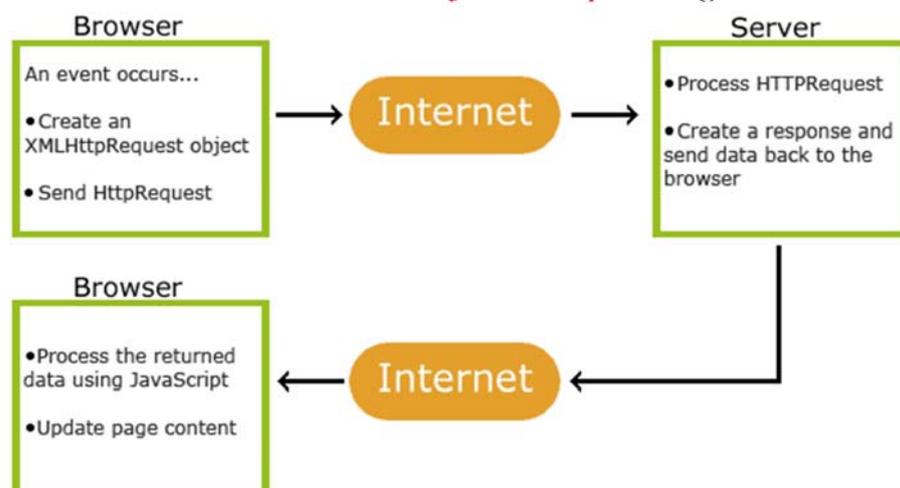
Le modèle Client-Serveur

- **Cloud** ou les environnements virtuels



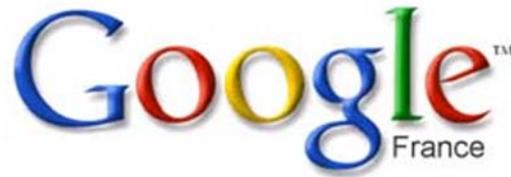
Le modèle Client-Serveur

- **Ajax** : (*Asynchronous JavaScript and XML*) est une technique de mise à jour des parties d'une page sans la charger en entier.
- Cela permet de gagner du temps en chargement et avoir des réponses quasi instantanément.
- La technique se base sur une méthode **JavaScript** intégrée aux navigateurs.



Le modèle Client-Serveur

- Application: *Google Suggest*



	marseille		Recherche avancée
			Préférences
			Outils linguistiques
Rech	marseille cassis	282 000 résultats	
	marseille liverpool	4 910 000 résultats	ance
	marseille foot	3 910 000 résultats	
	marseille 2013	759 000 résultats	
	marseille paris	1 070 000 résultats	
Programmes de	marseille provence	1 730 000 résultats	e.com in English
	marseille psg	365 000 résultats	
	marseille2night	1 430 résultats	
	marseille tourisme	1 210 000 résultats	
	marseille plus	762 000 résultats	
		fermer	

C'est quoi HTML 5.0 ?



- **HTML5** c'est avant tout le nouveau standard pour le langage HTML.
- **HTML5** est un standard en développement, mais déjà la majeure partie des navigateurs Web support la plupart de nouveaux éléments et APIs.
- Pour le nouveau **HTML5** certaines règles ont été établies:
 - Les nouvelles options doivent être basées sur HTML, CSS, DOM et JavaScript;
 - Réduire l'utilisation de plugins externes (Flash par exemple);
 - Améliorer les rapports d'erreurs;
 - Plus de balises fonctionnelles pour remplacer les scripts;
 - **HTML5** doit se rendre indépendant des unités d'utilisation (fixe et mobile)
 - Rendre transparent le processus de développement.
 - Les balises de base du HTML 4.01 restent opérationnelles.



Les nouveaux éléments

- Internet a beaucoup changé depuis que le HTML 4.01 est devenu standard en 1999.
- Aujourd'hui certains éléments du HTML 4.01 sont devenus obsolètes, d'autres n'ont jamais été utilisés, ou bien ils ont été appliqués dans des contextes différents de l'idées de leur création. Ces éléments sont retirés ou entièrement réécrits dans [HTML5](#).
- [HTML5](#) propose de nouveaux éléments sémantiques pour mieux structurer un document et fournir une meilleure manipulation de forme, dessin et contenu multimédia.
- Les nouveaux éléments sont repartis:
 - Nouveaux éléments sémantiques ou de structure
 - Nouveaux éléments multimédias
 - Nouveaux éléments de formulaire



Le document HTML5 minimal

```
<!doctype html>      <!-- Défini le type de document -->
<html lang="fr">     <!-- début avec langue par défaut -->
<head>               <!-- Entête du document -->
  <meta charset="utf-8"> <!-- Codage par défaut -->
  <title>Titre de la page</title>
  <!-- Feuille de style et script externe à importer -->
  <link rel="stylesheet" href="style.css">
  <script src="script.js">
  </script>
</head>
<body>               <!-- Corps du document -->
  ...
  <!-- Le contenu visualisé -->
  ...
</body>
</html>
```



Un document HTML5 minimal

- **`<!doctype html>`**
 - Son rôle est de préciser le type de document qui va suivre.
 - On permet ainsi au navigateur de savoir quel langage de la vaste famille SGML il devra interpréter.
- L'attribut **`lang`** de la balise **`<html>`**
 - **`<html lang="fr">`**
 - L'attribut *lang* précise la langue utilisée pour le contenu de la page, lorsqu'il est placé sur la racine `<html>`. Une langue bien indiquée sera utile ainsi aux synthèses vocales
- L'attribut **`charset`** de la balise **`<meta>`**
 - **`<meta charset="utf-8">`**
 - Le choix de l'UTF-8 est désormais préconisé par le W3C pour tous les protocoles échangeant du texte sur internet (dont HTML).



Un document HTML5 minimal

- L'élément **`<link>`**
 - **`<link rel="stylesheet" href="style.css">`**
 - Une balise `<link>` placée dans l'en-tête permet de mettre en relation la page avec d'autres documents externes, une feuilles de style CSS par exemple de type stylesheet.
- L'élément **`<script>`**
 - **`<script src="script.js"></script>`**
 - L'élément permet d'ajouter des scripts (JavaScript) qui vont s'exécuter dans le navigateur dès leur chargement.



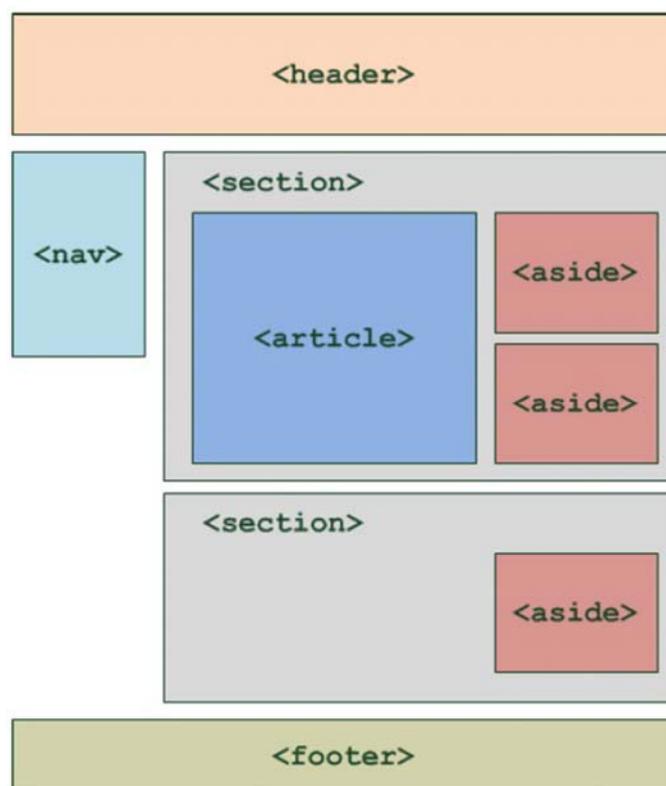
Un document HTML5 complet

- La partie `<body>` d'un document place tous les autres éléments HTML pour formater le contenu à travers: *paragraphes, listes, liens, images, vidéos, tableaux de données, formulaires, etc.*
- La majorité des éléments HTML sont toujours présents dans HTML5 complétés par de nouvelles balisages:
 - Éléments de section `<section>`, `<article>`, `<header>`, `<footer>`, `<nav>` et `<aside>`
 - Élément `<datalist>`
 - Éléments `<details>` et `<summary>`
 - Éléments `<figure>` et `<figcaption>`
 - Éléments média `<audio>` et `<video>` complétés par `<source>`
 - Élément `<canvas>`
 - Éléments `<meter>`, `<output>` et `<progress>`
 - Élément `<time>` et `<mark>`



Structure de la page Web HTML 5.0

-



Les éléments sémantiques et de structure



Balise	Description
<code><article></code>	Définit un article
<code><aside></code>	Définit le contenu mis à part le contenu de la page
<code><bdi></code>	Met une partie du texte isolé dans une direction différente du reste du texte
<code><command></code>	Définit un bouton de commande qu'un utilisateur peut invoquer
<code><details></code>	Définit des détails que l'utilisateur peut afficher ou masquer
<code><summary></code>	Définit un titre visible d'un élément <code><details></code>
<code><figure></code>	Spécifie le contenu autonome, comme les illustrations, schémas, photos, listes de codes, etc..
<code><figcaption></code>	Définit une légende pour un élément de <code><figure></code>
<code><footer></code>	Définit un pied de page d'un document ou une section
<code><header></code>	Définit l'en-tête d'un document ou section
<code><hgroup></code>	Regroupe un ensemble de <code><h1></code> à <code><h6></code> des éléments lorsqu'un titre a plusieurs niveaux

Les éléments sémantiques et de structure



Balise	Description
<code><mark></code>	Définit le texte marqué en surbrillance
<code><meter></code>	Définit une mesure scalaire dans une aire de répartition connue
<code><nav></code>	Définit les liens de navigation
<code><progress></code>	Représente la progression d'une tâche
<code><ruby></code>	Définit une annotation ruby (pour la typographie asiatique)
<code><rt></code>	Définit une explication/prononciation des caractères (pour la typographie asiatique)
<code><rp></code>	Définit ce qu'il faut afficher dans les navigateurs qui ne prennent pas en charge les annotations Rubys
<code><section></code>	Définit une section dans un document
<code><time></code>	Définit une date/heure
<code><wbr></code>	Définit un saut de ligne possible

Les nouveaux élément de formulaire



Balise	Description
<code><datalist></code>	Spécifie une liste de prédéfinis pour les options des contrôles d'entrée
<code><keygen></code>	Définit un champ générateur de clés (pour les formulaires)
<code><output></code>	Définit le résultat d'une opération

Les nouveaux éléments pour les médias



Balise	Description
<code><audio></code>	Définit un contenu audio – En savoir plus sur la lecture d'un fichier audio
<code><video></code>	Définit une vidéo ou un film
<code><source></code>	Définit de multiples ressources pour les médias <code><video></code> et <code><audio></code>
<code><embed></code>	Définit un conteneur pour une application externe ou un contenu interactif (un plug-in)
<code><track></code>	Définit des pistes de texte pour les médias <code><video></code> et <code><audio></code>

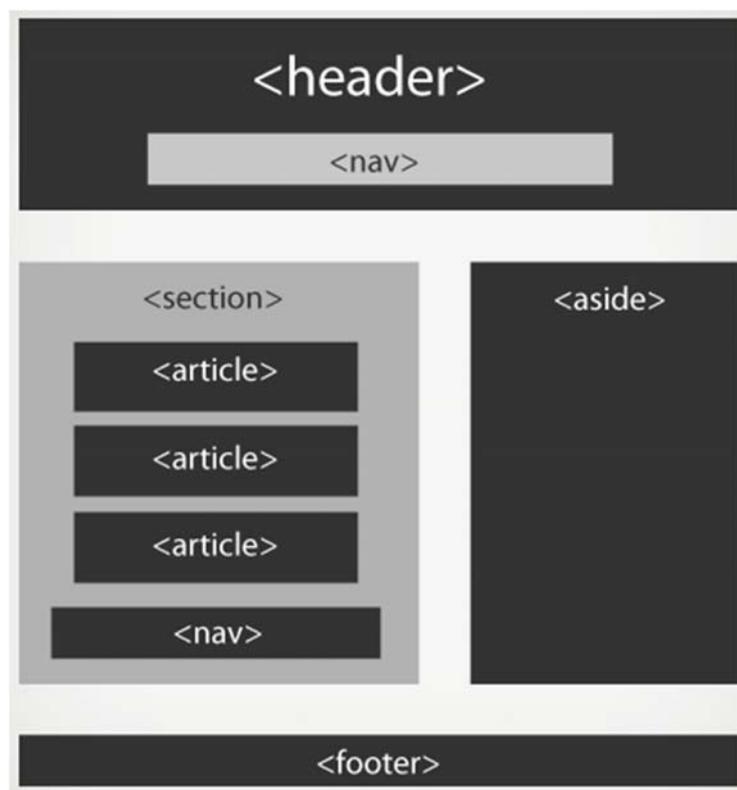


Les éléments structurants

- **HTML5** inclut la majorité des éléments **HTML4** pour assurer une rétrocompatibilité avec les navigateurs.
- Les éléments génériques comme `` ou `<div>` ont un rôle totalement neutre et ne servent qu'à regrouper d'autres éléments HTML pour leur affecter un style CSS commun, voire pour interagir avec eux via le DOM.
- Les éléments de section (`<section>`, `<article>`, `<nav>`, `<aside>`, `<header>`, `<footer>`) segmentent pour obtenir la structuration des portions du document ou de l'application Web, qui possèdent une valeur sémantique particulière.



Nouvelle structure d'une page Web





Les éléments structurants

- La partie entête d'un document en HTML4 se présente par la définition d'un CSS exprimé avec la balise `<div id="header">`
- Le nouvel élément `<header>` convient parfaitement à l'introduction d'un document.
- L'élément `<nav>` remplace avantageusement `<div id="menu">`.
- L'élément `<aside>` revêt le rôle de barre latérale
- L'élément `<section>` regroupe le contenu principal.
- Le `<footer>` est destiné au pied-de-page. Il peut accueillir des mentions spécifiques, un rappel du titre et du logo, des liens de navigation, etc.
- **Les éléments de section ne se limitent pas à un simple découpage et à une application de styles graphiques mais bien à une meilleure classification sémantique de l'information.**



CSS3 - feuilles de styles

Quatre méthodes pour intégrer les feuilles de styles en HTML5:

1. L'instruction `STYLE` en tant qu'élément :

```
<style type="text/css"> ... </style>
```

2. L'élément `LINK` pour établir un lien vers une feuille de style externe:

```
<link title="test" type="text/css" rel="stylesheet" href="ma_feuille.css">
```

3. L'instruction `STYLE` en tant qu'attribut:

```
<p style="font-size: 12pt; color: green">
```

4. Importer une feuille de style:

```
<style type="text/css">
```

```
  @import url(http://mon.site.fr/html/feuille.css);
```

```
</style>
```

Note : Le fichier `.css` ne doit pas contenir de code `HTML`



La mise en forme avec CSS3

- Une feuille de style CSS

```

balise1
{
propriete1: valeur1;
propriete2: valeur2;
}
balise2
{
propriete1: valeur1;
propriete2: valeur2;
propriete3: valeur3;
}
balise3
{
propriete1: valeur1;
}

```

Propriété	Description (exemple)
font	Propriété de police (family, size, ...)
text-align	left, center, right, justify
text-shadow	Ombre de texte (h, v, f, c)
color	Couleur du texte (nom, rgb)
background-color	Couleur de fond
padding-left	Marge intérieure à gauche
border-width	Épaisseur de bordure
margin-left	Marge à gauche
line-height	Hauteur de ligne
vertical-align	Alignement vertical (baseline, middle, sub, super, top, bottom)
word-wrap	Césure forcée
box-shadow	Ombre de boîte



CSS3 - feuilles de styles

```

<!doctype html><html><head>
<style type="text/css">
<!--
H1 { font-family: fantasy; font-size: 24pt; font-variant:
small-caps; color: green; text-align: center }
p { font-size: 10pt; color: black }
//-->
</style></head><body>
<h1>Le titre de cette page</h1>
<DIV STYLE="margin-left: 50px; color: blue">
Ce paragraphe sera placé à 40 pixels de la marge de gauche
et sera de couleur bleu.</div>
<p>Un paragraphe solitaire en 10 points.</p>
</body></html>

```

LE TITRE DE CETTE PAGE

Ce paragraphe sera placé à 40 pixels de la marge de gauche et sera de couleur bleu.

Un paragraphe solitaire en 10 points.

CSS3 - feuilles de styles



Propriété	Valeur(s)	Description	Exemple
font-size	xx-small, x-small, small, medium, large, x-large, xx-large, larger, smaller, ou taille défini en pixels (px) cm (cm), mm (mm), point (pt), pourcentage (%)	Taille de la police	{font-size: 12pt}
font-family	serif, sans-serif, cursive, fantasy, monospace, ou nom de la police (Arial, Verdana, etc...)	Type de police	{font-family: Arial}
font-weight	normal, bold, bolder, lighter, ou valeur numérique de 100 à 900 (par tranche de 100)	Poids de la police	{font-weight: bold}
font-style	normal, italic, oblique	Style de la police	{font-weight: italic}
font-variant	normal, small-caps	Variante de la police	{font-weight: small-caps}
line-height	point (pt), pourcentage (%)	Hauteur de la ligne	{line-height: 5pt}
color	nom ou valeur hexadécimale de la couleur	Couleur de la police	{color: green}
background-image	URL du fichier	Image d'arrière-plan	{background-image: url(image/fichier.gif)}
background-repeat	repeat, no-repeat, repeat-x, repeat-y	Répétition de l'arrière-plan	{background-repeat: no-repeat}
background-attachment	scroll, fixed	Effet de filigrane	{background-attachment: fixed}
background-position	top, center, bottom, left, right	Position de l'arrière-plan (verticale et horizontale)	{background-position: center center}
vertical-align	baseline, sub, super, top, text-top, middle, bottom, text-bottom, ou valeurs en pourcentage	Alignement vertical	{vertical-align: top}

CSS3 - feuilles de styles



Propriété	Valeur(s)	Description	Exemple
text-align	left, right, center, justify	Alignement du texte	{text-align: justify}
text-indent	valeur en pixel ou pourcentage	Retrait de la première ligne	{text-indent: 10px}
text-decoration	none, underline, overline, line-through	Décoration du texte	{text-decoration: overline}
text-transform	capitalize, uppercase, lowercase, none	Casse du texte	{text-transform: none}
margin-left (ou -right, -top, -bottom)	valeur en pixel ou pourcentage	Valeurs des marges	{margin-right: 10px}
padding-left (ou -right, -top, -bottom)	valeur en pixel ou pourcentage	Remplissage du bloc	{padding-left: 15pt}
border-left (ou -right, -top, -bottom)	medium, thick, ou un style	Type de bordure	{border-top: medium}
border-x-width (x à remplacer par right, left, top, bottom)	medium, thick ou taille.	Taille de la bordure	{border-top-width: 15}
border-style	none, dotted, dashed, solid, double, groove, ridge, inset, outset	Style de la bordure	{border-style: ridge}
border-color	nom ou valeur hexadécimale de la couleur	Couleur de la bordure	{border-color: red}
float	none, left, right, both	Alignement d'un bloc par rapport à un autre	{float: right}
clear	none, left, right, both	Autorise l'alignement d'un bloc par rapport à un autre	{clear: right}
letter-spacing	normal ou valeur en point	Espacement des lettres	{letter-spacing: 2pt}

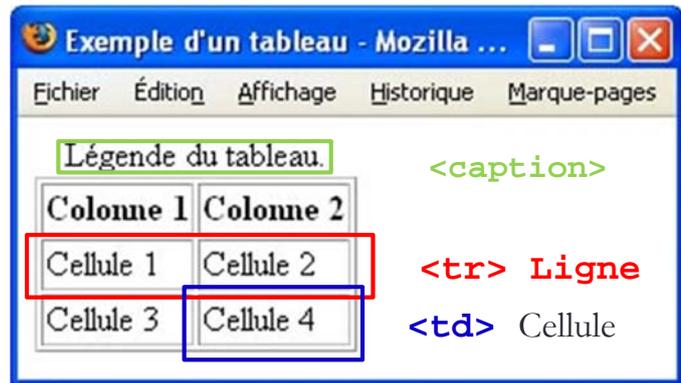
Les éléments HTML5 de base (Table)



```

<!doctype html>
<HTML>
<HEAD><TITLE>Tableau</TITLE>
</HEAD>
<BODY>
  <TABLE BORDER="1">
    <CAPTION> Légende </CAPTION>
    <TR>
      <TH>Colonne 1</TH>
      <TH>Colonne 2</TH>
    </TR>
    <TR>
      <TD>Cellule 1</TD>
      <TD>Cellule 2</TD>
    </TR>
    <TR>
      <TD>Cellule 3</TD>
      <TD>Cellule 4</TD>
    </TR>
  </TABLE>
</BODY>
</HTML>

```



Les éléments HTML5 de base (Table)



- Fusionner les lignes et/ou les colonnes:

```

<!doctype html><HTML><HEAD>
<TITLE>Exemple d'un tableau</TITLE></HEAD>
<BODY>
  <TABLE BORDER="1">
    <TR><TD>Cellule 1</TD>
    <TD COLSPAN="2">2 cellules fusionnées (sur la même ligne)</TD>
  </TR>
  <TR><TD>Cellule 2</TD><TD>Cellule 3</TD>
  <TD ROWSPAN="2">2 cellules fusionnées (sur la même
  colonne)</TD>
  </TR>
  <TR>
    <TD>Cellule 4</TD>
    <TD>Cellule 5</TD>
  </TR>
  </TABLE>
</BODY>
</HTML>

```



HTML5 - exemple tableau



```

<!doctype html><html lang="fr"><head><meta charset="utf-8">
  <link rel="stylesheet" href="style.css" />
  <title>Annuaire t&eacute;l&eacute;phonique</title></head>
<body>
<table border="0" align="center" cellpadding="2" cellspacing="3" bgcolor="#33cc66"
width="350">
<caption>Annuaire t&eacute;l&eacute;phonique</caption>
<thead> <!-- En-tête du tableau -->
  <tr>
    <th>Nom</th>
    <th>T&eacute;l&eacute;phone</th>
    <th>Poste</th>
  </tr>
</thead>
<tfoot> <!-- Pied de tableau -->
  <tr>
    <th>Nom</th>
    <th>T&eacute;l&eacute;phone</th>
    <th>Poste</th>
  </tr>
</tfoot>
<tbody> <!-- Corps du tableau -->
<tr align="center"><td class="nom"> Martin A. </td><td class="tel"> 04 76 98 45 34 </td><td
class="pst"> 101 </td></tr>
<tr align="center"><td class="nom"> Dupont B. </td><td class="tel"> 04 72 98 45 34 </td><td
class="pst"> 102 </td></tr>
<tr align="center"><td class="nom"> Durand J. </td><td class="tel"> 04 74 98 45 34 </td><td
class="pst"> 103 </td></tr>
</tbody></table></body></html>

```

Annuaire téléphonique

Nom	Téléphone	Poste
Martin A.	04 76 98 45 34	101
Dupont B.	04 72 98 45 34	102
Durand J.	04 74 98 45 34	103
Nom	Téléphone	Poste

HTML5 - exemple CSS3 (style.css)



```

td {
border: 1px solid black; /* auront une bordure de 1px */
}
td.nom /* les cellules des noms */
{
background-color: #FF0000;
color: #0000CD;
font-weight: bold; }
td.tel /* les cellules des tel */
{
background-color: blue;
color: yellow;
font-style: italic;
}
td.pst /* les cellules des postes */
{
background-color: LightGrey;
color: black;
}
caption /* titre du tableau */
{
font-family: Times;
font-size: 200%;
text-shadow: 5px 5px 5px #0000CD; /* CSS3 */
}

```



Les éléments HTML5: <Form>

- Le formulaire (**<form>**) est l'élément **HTML5** interactif présenté côté client pour l'envoi des informations côté serveur.
- Côté serveur les données reçues sont traitées par un langage de script.
- Un formulaire propose les fonctionnalités suivantes:
 - saisir du texte,
 - sélectionner des options par les cases à cocher et les listes déroulantes
 - valider avec un bouton
 - activer des scripts (JavaScript) pour valider les saisies
- Un formulaire envoie les données saisies ou sélectionnées par l'**action** en précisant la méthode d'envoi:
 - **GET** méthode peu adaptée car limitée à 255 caractères.
 - **POST** est la méthode la plus utilisée pour les formulaires car elle permet d'envoyer un grand nombre d'informations à l'adresse de l'**action**.



Les éléments HTML5: <Form>

Balise	Description
<form>... </form>	Définir le formulaire
<input>	Définir un champ de saisie
<textarea>	Définir un champ de saisie multi-lignes
<label>	Etiquette pour un champ de saisie <input>
<fieldset>	Regrouper des éléments liés dans un formulaire
<legend>	Définir le titre pour la balise <fieldset>
<select>	Définir une liste déroulante
<optgroup>	Définir un groupe d'options dans une liste déroulante
<option>	Définir un éléments dans une liste déroulante
<button>	Définir l'objet bouton
<datalist>	Définir une liste d'options prédéfinies pour une saisie
<keygen>	Définir un jeu de clés pour le cryptage et le décryptage
<output>	Représente la somme d'un calcul. Attribut: <i>for, name, et form</i>

Les éléments HTML5 : Formulaire



```
<form method="post" action="etudes.php">
<p>Etudiant IUT-R&T</p><p><label>Nom: </label><input type="text" name="nom"></p>
<p><label>Prénom: </label>
<input type="text" name="prenom"></p>
<p><label>Age: </label>
<input type="text" name="age"></p>
<p>Année d'étude:<br />
<label>Première: </label>
<input type="radio" name="annee" value="1">
<label>Deuxième: </label>
<input type="radio" name="annee" value="2">
<label>Licence: </label>
<input type="radio" name="annee" value="3"></p>
<p><label>Matière: </label>
<select name="matiere">
<option value="Math">Math</option>
<option value="Info">Info</option>
<option value="WLAN">WLAN</option>
<option value="Comm">Comm</option>
<option value="TCom">TeleC</option>
</select></p>
<p><label>Moyenne: </label><input type="text" name="moyenne" value="0.0"></p>
<p><label>Soumettre: </label><input type="submit" value="Envoyer"></p></form>
```

Etudiant IUT-R&T

Nom:

Prénom:

Age:

Année d'étude

Première: Deuxième: Licence:

Matière:

Moyenne:

Soumettre:

Les éléments HTML5 <Form>



```
<form method="post" action="traiter.php">
<p>
<label for="pseudo">Votre pseudo :</label>
<input type="text" name="pseudo" id="pseudo"
placeholder="Ex: Zoro" size="30" maxlength="10"/>
</p>
<p>
<label for="pass">Votre mot de passe :</label>
<input type="password" name="pass" id="pass" />
</p>
</form>
```

Pour lier LABEL
au INPUT:
for="nom"
id="nom"

placeholder: texte exemple
size: la taille du champ
maxlength: nombre de
caractères

name: nom de variable
id: identificateur local

champ type "password"
masque la saisie

Votre pseudo :

Votre mot de passe :

HTML5 <Form> : saisie enrichies



- HTML5 apporte des fonctionnalités nouvelles relatives aux types *input*.
- *Attention!* Tous les navigateurs ne reconnaissent pas ces types de saisie enrichies

- `<input type="email" />`

- Soit: `abc@iut.fr`

- `<input type="url" />`

- Soit: `http://abc.iut.fr`

- `<input type="tel" />`

- Soit la saisie d'un numéro de téléphone

- `<input type="number" />`

- Soit un champ qui permet de saisir un nombre entier

- `<input type="date" />` /* pour: date, time, week, month */

- Pour l'instant peu de navigateurs gèrent ce type de champ à part Opera

HTML5 <Form> : contrôle par pattern



- Supporter par: Firefox 4b7, Chrome 6, Opera 9, Safari 5.0.3

Description	Pattern
Numéro de carte de crédit	<code>[0-9]{13,16}</code>
Alphanumérique	<code>[a-zA-Z0-9]+</code>
Lettres	<code>[a-zA-Z]+</code>
Adresse IPv4	<code>((^ \.)((25[0-5]) (2[0-4]\d) (\d\d) ([1-9]?\d)))\{4\}\$</code>
Adresse IPv6	<code>((^ :)([0-9a-fA-F]{0,4}))\{1,8\}\$</code>
Date (MM/DD/YYYY)	<code>(0[1-9] 1[012])[- /.](0[1-9] 12)[0-9] 3[01])[- /.](19 20)\d\d</code>
Numéro de téléphone (+99(99)9999-9999)	<code>[\+]\d{2}[\(\)\d{2}[\)]\d{4}[\-]\d{4}</code>
Prix (Format: 1,00)	<code>\d+(,\d{2})?</code>



Expression régulière

- Les expressions régulières sont des modèles créés à l'aide de caractères ASCII permettant de manipuler des chaînes de caractères.
 - Ils permettent notamment de trouver les portions de la chaîne de caractères correspondant au modèle.
- C'est un système ingénieux et puissant permettant de retrouver un mot, un chiffre, un nombre ou une phrase dans un texte, ressemblant au modèle que l'on a construit.
- Ainsi, les expressions régulières permettent de rechercher des occurrences (c'est-à-dire une suite de caractères correspondant à ce que l'on recherche) grâce à une série de caractères spéciaux. L'expression régulière en elle-même est donc une chaîne de caractère contenant des caractères spéciaux et des caractères standards.



Expression régulière

- Les symboles `^` et `$` indiquent le début ou la fin d'une chaîne, et permettent donc de la délimiter.
- Les symboles `*`, `+` et `?`, respectivement "zero ou plusieurs", "un ou plusieurs", "un ou aucun", permettent de donner une notions de nombre.
- Les accolades `{X,Y}` permettent de donner des limites de nombre.
- Les parenthèses `()` permettent de représenter une séquence de caractères.
- La barre verticale `|` se comporte en tant qu'opérateur OU
- Le point `.` indique n'importe quel caractère (une fois)
- Les crochets `[]` définissent une liste de caractères autorisés (ou interdits). Le signe `-` permet quand à lui de définir un intervalle. Le caractère `^` après le premier crochet indique quand à lui une interdiction.



HTML5: les hyperliens

- Liens internes (même page)
 - Définir la cible d'un lien (l'ancre):
``
 - Associer un lien à une ancre:
`top`
- Liens internes (page différente)
 - `Cliquez ici`
 - `Cliquez ici`
- Liens externes (A HREF=" ... ")
 - `Découvrez Yahoo France`



HTML5: les images

- Pour insérer une image dans un document HTML il faut utiliser la balise ****:

```
<IMG SRC="gif/livres.gif"  
WIDTH="33" HEIGHT="34"  
ALT="Mes livres"  
border="0">
```



- Trois formats graphiques reconnus sur le Web
 - GIF (*Graphics Interchange Format*)
 - JPEG (*Joint Photographic Experts Group*)
 - PNG (*Portable Network Graphic*)



HTML5: les images

- Le Format **GIF** est un format de fichier graphique **bitmap** (raster).
 - Maximum de 256 couleurs;
 - Option transparence (une couleur);
 - Option d'entrelacement qui permet d'afficher l'image progressivement;
 - Animation
- Le format forme des images de petite taille bien adaptées au Web pour la présentation:

- Logo
- Image animée
- Schéma
- Présentation graphique



HTML5: les images

- Le Format **JPEG** est un format de fichier graphique **bitmap** (raster).
 - Qualité 16.5 millions de couleurs;
 - Taux de compression de 1 à 99%;
 - Ne gère pas l'effet de transparence;
 - Affichage progressif (*interlaced*).
- Présentation en qualité photographique des images.
- Volume des fichiers souvent important





HTML5: les images

- Le Format **PNG**

- compression sans perte de données;
- une palette indexée jusqu'à 256 couleurs, niveaux de gris jusqu'à 16 bits et les couleurs réelles jusqu'à 42 bits;
- transparence de 254 niveaux;
- méthode d'entrelacement très performante;
- ne permet pas de créer des animations;
- moins performant que le JPEG



HTML5: Les formats audio

- Pour diffuser de la musique ou des sons de nombreux formats sont disponibles.
 - **MP3** : C'est le format le plus compatible. Tous les appareils et navigateurs savent lire des MP3.
 - **AAC** : Ce format est utilisé par Apple sur iTunes. C'est un format de bonne qualité. Lu par les iPod et iPhone.
 - **OGG** : Le format *Ogg Vorbis* est très répandu dans le monde du logiciel libre, notamment sous Linux. Ce format a l'avantage d'être libre.
 - **WAV** : C'est un format audio non compressé utilisé par les anciens logiciels Microsoft. Il est à éviter autant que possible de l'utiliser car le fichier est très volumineux avec ce format.



HTML5: Les formats vidéo

- Le stockage de la vidéo dépend de trois éléments :
 - Un **format conteneur** : c'est la boîte qui sert à contenir les deux éléments audio et vidéo. Le type de conteneur est reconnu à l'extension du fichier : **AVI, MP4, MKV...**
 - Un **codec audio** : c'est le format du son de la vidéo, généralement compressé: **MP3, AAC, OGG, ...**
 - Un **codec vidéo** : c'est le format qui compresse les séquences d'images. Les principaux pour le Web sont :
 - **H.264** : l'un des plus puissants et des plus utilisés, mais il n'est pas entièrement gratuit.
 - **Ogg Theora** : un codec gratuit et libre de droits, mais moins puissant que H.264. Il est bien reconnu sous Linux.
 - **WebM** : un autre codec gratuit et libre de droits, plus récent. Proposé par Google, c'est un concurrent le direct du format H.264.



HTML5: insertion d'un fichier audio

- La balise **<audio>** et ses attributs:
 - **<audio src="musique.mp3"></audio>**
 - **controls** : ajoute les boutons "Lecture", "Pause" et la barre de défilement.
 - **width** : pour modifier la largeur de l'outil de lecture audio.
 - **loop** : la musique sera jouée en boucle.
 - **autoplay** : la musique sera jouée dès le chargement de la page.
 - **preload** : indique si la musique peut être pré-chargée dès le chargement de la page ou non. Cet attribut peut prendre les valeurs :
 - **auto** (par défaut) : le navigateur décide s'il doit pré-charger toute la musique,
 - **metadata** : charge uniquement les métadonnées,
 - **none** : pas de pré-chargement.





HTML5: insertion d'une vidéo

- `<video src="zoro.mp4">`
 - **poster** : image à afficher à la place de la vidéo tant qu'elle n'est pas encore lancée.
 - **controls** : pour ajouter les boutons "Lecture", "Pause" et la barre de défilement.
 - **width** : pour modifier la largeur de la vidéo.
 - **height** : pour modifier la hauteur de la vidéo.
 - **loop** : la vidéo sera jouée en boucle.
 - **autoplay** : la vidéo sera jouée dès le chargement de la page.
 - **preload** : indique si la vidéo peut être pré-chargée dès le chargement de la page ou non suivant les attributs à préciser:
 - **auto** (par défaut) : le navigateur décide s'il doit pré-charger toute la vidéo.
 - **metadata** : charge uniquement les métadonnées
 - **none** : pas de pré-chargement.



HTML5: le site Web

The screenshot shows a website layout with a light blue header containing the logo 'réseaux & iut télécoms' and the title 'Les Neticiens'. Below the header is a light green main content area. On the left side of this area is a light blue sidebar with a list of links: 'Accueil', 'Enseignants', and 'Etudiants'. The main content area contains two sections: 'À propos de la formation' with a paragraph about training in 'Réseaux et Télécommunications' and 'La vie étudiante' with the text 'Loisirs, études, études loisirs'. At the bottom is a yellow footer with the text 'Copyright IUT-R&T, Tous droits réservés' and a link 'Nous contacter!'.

HTML5: le code du site Web



```
<!DOCTYPE html><html><head><meta charset="utf-8" />
<link rel="stylesheet" href="sitestyle.css">
<title>Le Site Web</title></head><body><header>
<p><IMG SRC="logo-neticiens.gif" WIDTH="418" HEIGHT="78"
ALT="Neticiens" border="0"></p><h1>Les Neticiens</h1></header>
<nav><ul>
<li><a href="#">Accueil</a></li>
<li><a href="#">Enseignants</a></li>
<li><a href="#">Etudiants</a></li>
</ul></nav>
<section><aside>
<h1>Aacute; propos de la formation</h1>
<p>La formation en "R&eacute;seaux et
T&eacute;l&eacute;communications" <br />a pour avantage de promouvoir
les techniques et <br />les technologies du Web et du monde mobile
<br />des GSM.</p>
</aside><article><h1>La vie &eacute;tudiante</h1>
<p>Loisirs, &eacute;tudes, &eacute;tudes loisirs</p>
</article></section><footer>
<p>Copyright IUT-R&T, Tous droits r&eacute;serv&eacute;s<br />
<a href="#">Nous contacter!</a></p></footer>
</body></html>
```

HTML5: le CSS3 du site Web



```
header
{ text-align: center;
text-shadow: 4px 4px 2px #A9A9A9;
color: #EF7800;
width: 520px;
border: 1px solid black;
background-color: #E6E6FA; }
nav
{ float: left;
width: 169px;
border: 1px solid red;
background-color: #AFEEEE; }
section
{ margin-left: 171px;
padding-right: 2px;
width: 347px;
border: 1px solid blue;
background-color: #90EE90; }
footer { text-align: center;
width: 520px;
border: 1px solid black;
background-color: #F0E68C; }
```

HTML5: le cours en ligne



<http://139.124.26.245/pi>

<http://ivmad.free.fr/pi>