



HTML

# Généralités



# Généralités

## + A consulter

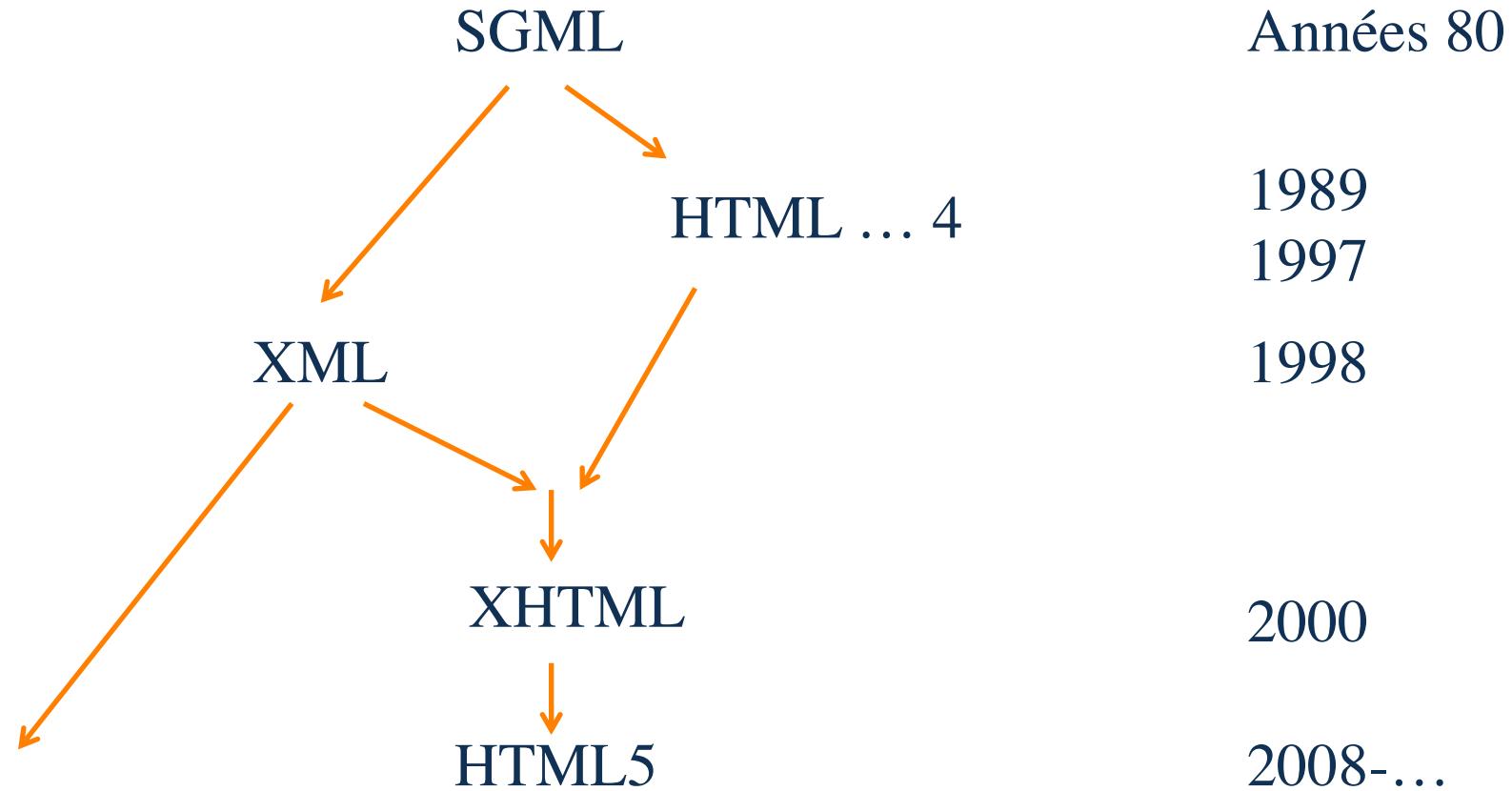
- + <http://www.w3c.org>
- + <http://41mag.fr/apprendre-le-html5-tutoriel-complet>
- + <http://validator.w3.org>
- + <http://www.w3schools.com>

# Généralités

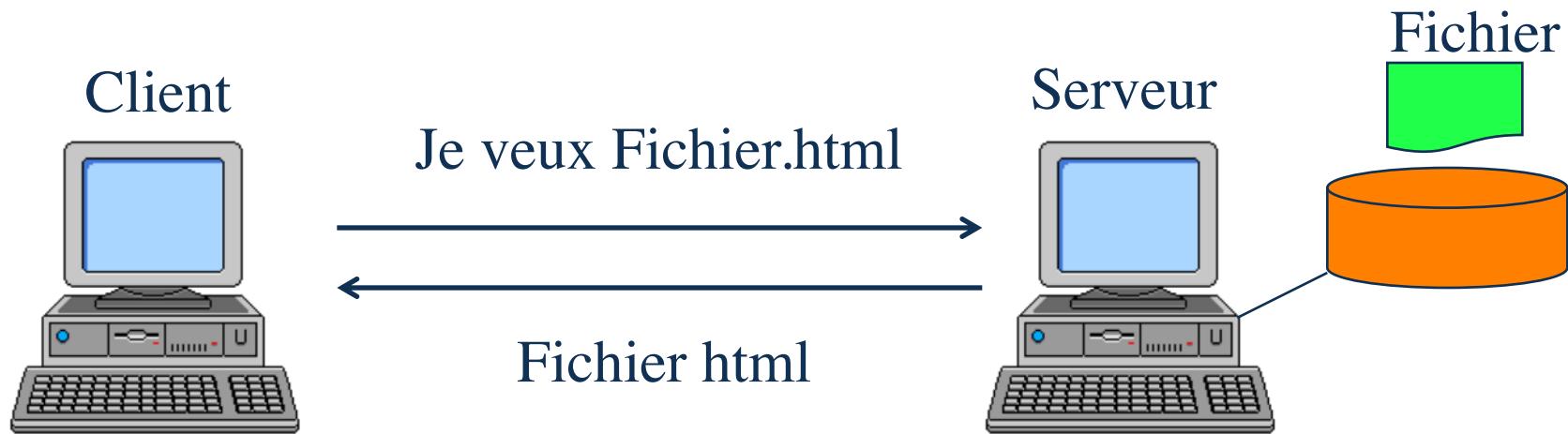
- + HTML = Langage de publication des pages web
  - + Définit le contenu et la forme d' un document web
  - + Né en 1991 au CERN
- + Dérivé du langage SGML

# Généralités

## + Historique des structures HTML

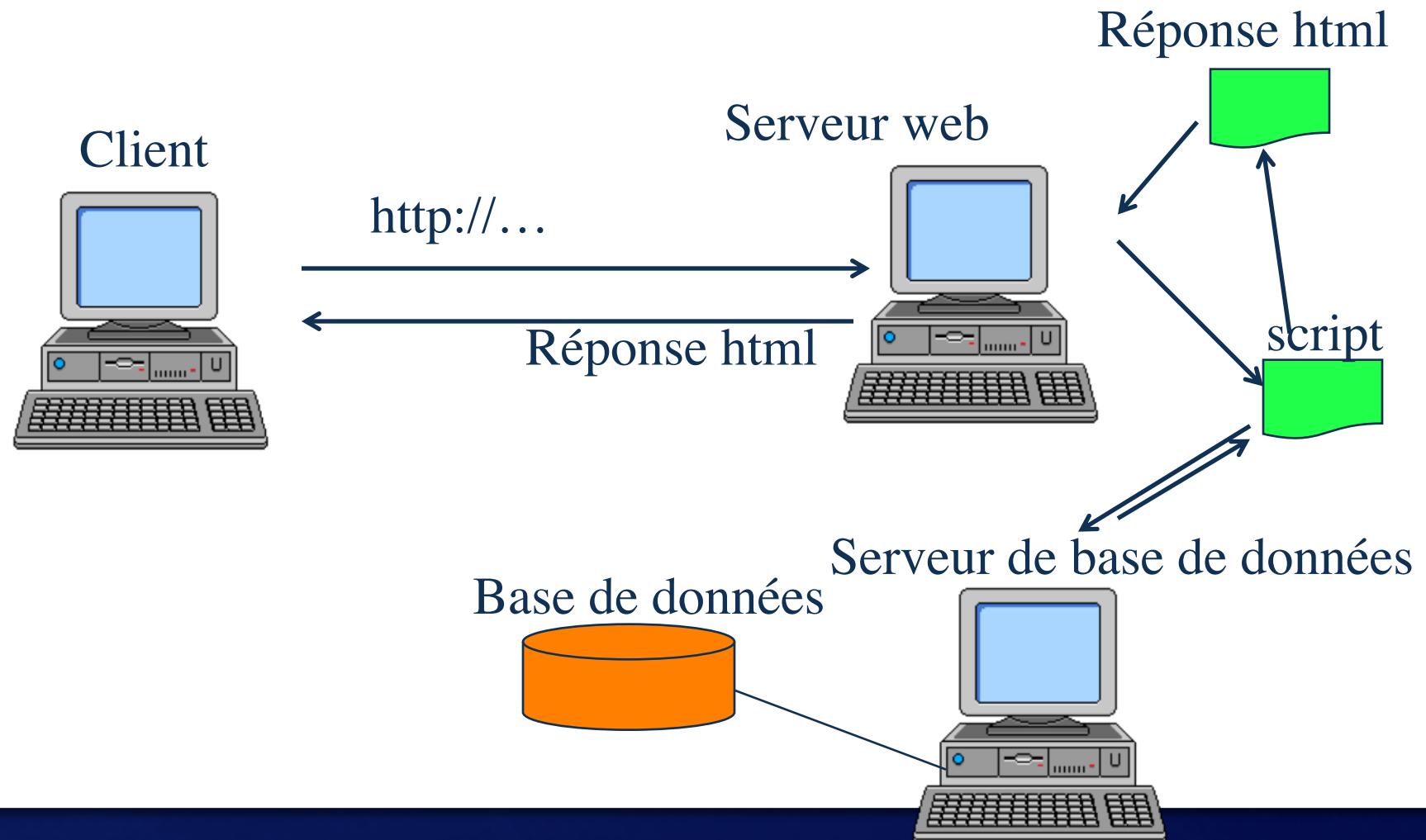


# Généralités



Récupération du fichier html  
Décodage des instructions  
Affichage des informations

# Généralités



# Généralités

- + HTML : texte
  - => peut être vu / créé à partir de n'importe quel éditeur de texte, sur n'importe quel système d'exploitation
- + Suit une syntaxe particulière : Langage à base de balises (héritage de SGML)
- + Ne fonctionne pas strictement de la même manière sur tous les navigateurs
  - => tester

# Généralités

## + Comment écrire du HTML

- + Editeurs - Logiciels payants  
Dreamweaver, Axure, ...  
Word
- + Editeurs gratuits  
NVU, Komposer, openElement, ...
- + Editeur de texte  
Notepad, Wordpad, Atom, Sublime Texte, Eclipse, Netbeans,  
BBEdit, ...

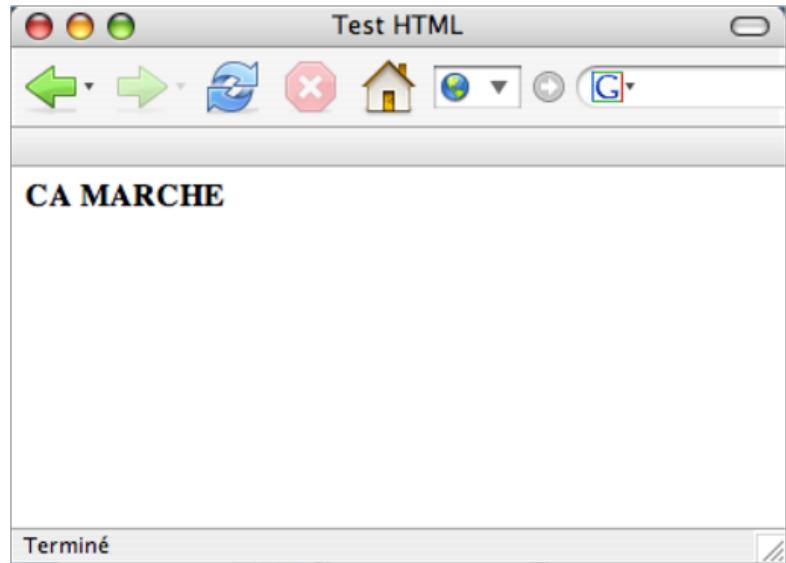
# HTML - Structure



# HTML

## + Une page HTML

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
  </head>
  <body>
    <p><strong>CA MARCHE</strong></p>
  </body>
</html>
```



# HTML

- + Structure d'une page

- + Spécifications

- + Titre / entête

- + Corps

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
  </head>
  <body>
    <p><strong>CA MARCHE</strong></p>
  </body>
</html>
```

# HTML

- + Un fichier HTML : maPage.html
  - + Comporte un nom **sans accent ni espace**
  - + Comporte une extension **.html**
- + Contient du texte visualisable par n'importe quel éditeur de texte, sur n'importe quelle type d'ordinateur
- + La page appelée par défaut sur un site est **index.html ou index.htm ou default.html**  
Information enregistrée dans le fichier de configuration du serveur web.

# HTML

- + Le texte d'un fichier HTML comporte des **balises** qui permettent de définir les **entités** de la page et de préciser leur mise en forme.
  - + Balise : suite de caractères entre <>.
    - + <html>
    - + <p>
    - + <div>
    - + 
    - + ...
  - + Seules les balises HTML peuvent être utilisées

# HTML

## + Quelques règles de base

- + Sauf exception, une balise ouverte doit être fermée.**

<XXX> ... </XXX>

→ →

ouverture

# fermeture

- + Une balise vide peut s'écrire**

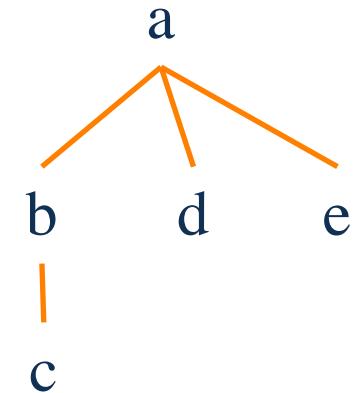
<XXX> </XXX> ou <XXX />

`<xxx/>` représente l'ouverture ET la fermeture de la balise

# HTML

- + Quelques règles de base
  - + Une balise peut contenir d'autres balises.  
Cela crée un arbre n-aire.

```
<a>
  <b>
    <c> ... </c>
  </b>
  <d> ... </d>
<e />
</a>
```



- + Les balises imbriquées ne doivent pas se croiser.  
`<a><b></a></b>` NON !  
`<a><b></b></a>`
- + Espace, CR, TAB = espace

# HTML

- + Une balise peut comporter des attributs :  
`<nom_balise attributs>....</nom_balise>`
  - + nom\_balise : identifie les informations et la façon de les afficher. Ce nom ne comporte ni accent, ni espace. Il ne commence pas par un chiffre.
  - + attributs : précision sur la balise  
nom\_attribut="valeur\_attribut"

Pas d'espace autour du =  
La valeur de l'attribut est entre guillemets

- + Exemple : `<div id="mainContent" class="content" > ... </div>`

# HTML

## + Aspect général d'un fichier html 5

```
<!DOCTYPE html>
<html lang="fr-fr">
    <head>
        entête
    </head>
    <body>
        corps
    </body>
</html>
```



# HTML

- + Le contenu d'un fichier html
  - + DOCTYPE indique un type de document XML suivant une syntaxe particulière
  - + Une balise générale : <html> ... </html>
  - + Une balise entête (head) :
    - + Titre du document
    - + Informations sur le document, sa mise en page, ...
  - + Une balise corps (body) :
    - + Contenu du document
    - + Informations à afficher, de quelle manière les afficher, ...

# HTML - Entête

## + La partie entête : head

- + Informations sur la page (pas affiché)
- + Dans la balise d'entête : <head> ... </head>
- + Contient d'autre balises :
  - + Titre : <title>ceci est un titre</title>
  - + Propriétés du document <meta name ="xx" content = "yy">  
Auteur, Mots clés, encodage, ...
  - + Fichiers CSS: <link ...>
  - + Fichiers JS: <script src= »...» type="text/javascript"></script>

# HTML - Entête

## + Entête : exemple

```
<head>
    <title>Page de garde</title>
    <meta charset="UTF-8" />
    <meta name="Author" content="Centrale" />
    <meta name="Description" content="Page de garde" />
    <link href="Pages.css" rel="stylesheet" type="text/css" />
    <script src="js/bootstrap.min.js" type="text/javascript"></script>
</head>
```

# HTML - Entête

- + A quoi sert une feuille de style ?  
Feuille de style = CSS (Cascade Style Sheet)
  - + Définir la façon d'afficher certaines balises
  - + Harmoniser l'affichage des balises dans une page ou d'une page à l'autre

```
<html>
  <head>
    ...
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
    ...
  </head>
  <body>
    <a href=...> <... </a>
  </body>
</html>
```

Page.html

```
a {
  ...
}

td {
  ...
}
```

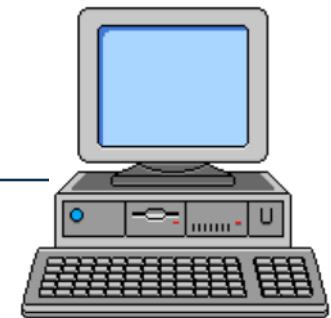
Feuille.css

# HTML - Entête

## + Feuille de style



Navigateur :



Serveur :

Je veux Page.html

Envoi Page.html

Je veux Feuille.css

Envoi Feuille.css

Analyse

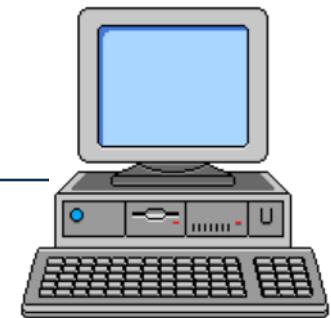
Affichage

# HTML - Entête

## + Feuille de style – Utilisation du cache



Navigateur :



Serveur :

Je veux Page2.html

Envoi Page2.html

Analyse

*Je connais Feuille.css => je ne le redemande pas*

Affichage

# HTML - Corps

## + Corps du document

<body ....>

*contenu= (texte + balises de mise en forme + ...)*

</body>

# HTML - Corps

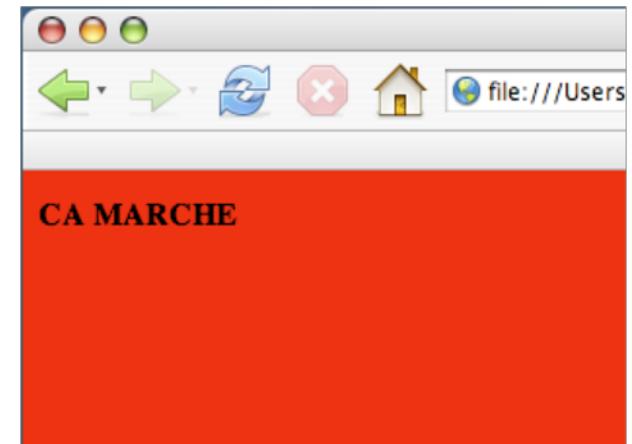
## + ID, Classes et style

- + Attribut ID : permet de définir un identifiant (unique pour la page)
- + Attribut class : permet d'indiquer que la balise appartient à un groupe de balise (la classe)
- + Attribut style : permet de définir le style à appliquer sur la balise qui le comporte

# HTML - Corps

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
  </head>

  <body style="background-color:red">
    <p><strong>CA MARCHE</strong></p>
  </body>
</html>
```

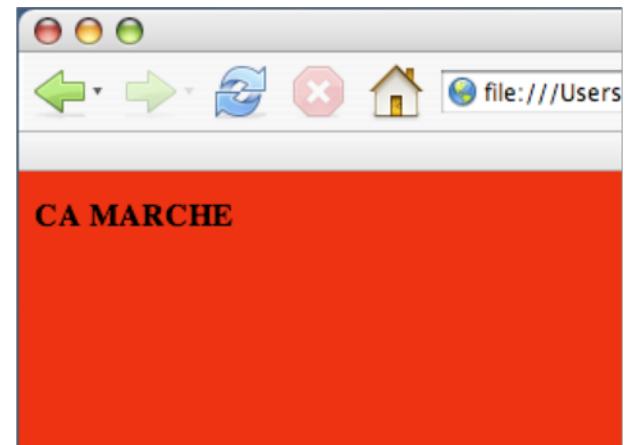


# HTML - Corps

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body>
    <p><strong>CA MARCHE</strong></p>
  </body>
</html>
```

```
body {
  background-color:red;
}
```

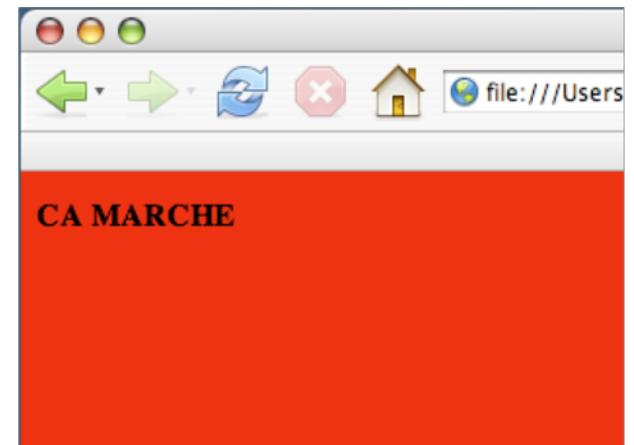


# HTML - Corps

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body class="background">
    <p><strong>CA MARCHE</strong></p>
  </body>
</html>
```

```
.background {
  background-color:red;
}
```

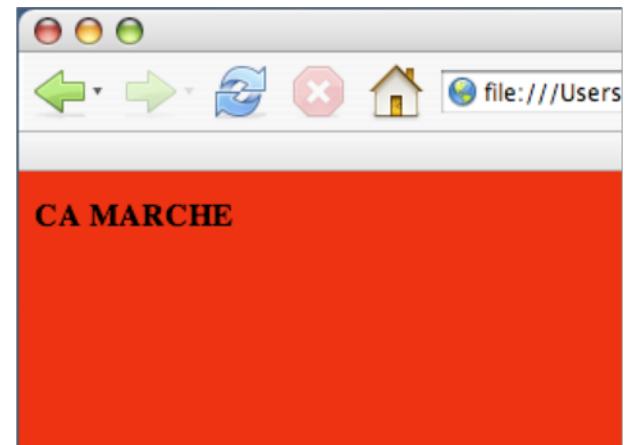


# HTML - Corps

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body id="background">
    <p><strong>CA MARCHE</strong></p>
  </body>
</html>
```

```
#background {
  background-color:red;
}
```



# HTML - Corps

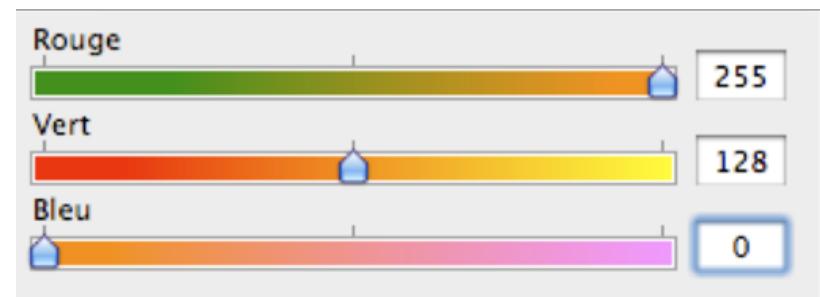
## + Les couleurs :

### + Couleurs nommées :

red, green, blue, purple, pink, white, black, orange, yellow, ...

### + Couleurs en RGB :

rgb( 0..255, 0..255, 0..255 )



### + Couleur en hexadécimal :

#rrggbb où rr, gg, bb sont des valeurs entre 00 et FF.  
#rgb où r,g,b sont des valeurs entre 0 et f.

# HTML - Corps

- + Quelques couleurs :
  - + Blanc : #FFFFFF
  - + Noir : #oooooooo
  - + Rouge : #FFoooo
  - + Vert : #ooFFoo
  - + Bleu : #ooooFF
  - + Jaune : #FFFFoo
  - + Rose: #FFooFF
  - + Cyan : #ooFFFF
  - + Orange : #FF8000
  - + Violet : #8000FF

Code hexa	Couleur	Code hexa	Couleur	Code hexa	Couleur
#FFFFFF		#CCFFFF		#99FFFF	
#FFFFCC		#CCFFCC		#99FFCC	
#FFFF99		#CCFF99		#99FF99	
#FFFF66		#CCFF66		#99FF66	
#FFFF33		#CCFF33		#99FF33	
#FFFF00		#CCFF00		#99FF00	
#FFCCFF		#CCCCFF		#99CCFF	
#FFCCCC		#CCCCCC		#99CCCC	
#FFCC99		#CCCC99		#99CC99	
#FFCC66		#CCCC66		#99CC66	
#FFCC33		#CCCC33		#99CC33	
#FFCC00		#CCCC00		#99CC00	
#FF99FF		#CC99FF		#9999FF	
#FF99CC		#CC99CC		#9999CC	
#FF9999		#CC9999		#999999	
#FF9966		#CC9966		#999966	
#FF9933		#CC9933		#999933	
#FF9900		#CC9900		#999900	
#FF66FF		#CC66FF		#9966FF	
#FF66CC		#CC66CC		#9966CC	
#FF6699		#CC6699		#996699	
#FF6666		#CC6666		#996666	
#FF6633		#CC6633		#996633	

# HTML - Corps

- + Prise en compte des styles
  - + Les styles sont appliqués dans l'ordre où ils apparaissent
  - + Un style peut écraser partiellement un autre
- + Mais il y a aussi des priorités :
  - + Les styles internes (définis dans la balise) sont appliqués après les styles externes (dans une css)
  - + Les styles des id sont appliqués après les styles sur les classes

# HTML – Les balises du corps



# HTML - Elements

- + Que contient une page html ?
  - + Blocs
  - + Titres
  - + Paragraphes
  - + Tableaux
  - + Listes
  - + ...

# HTML - Elements

## + Elements de texte

### + Balise de paragraphe

```
<p>Ceci est un texte</p>
```

### + Balises d'entêtes

```
<h1>...</h1>
```

```
<h2>...</h2>
```

...

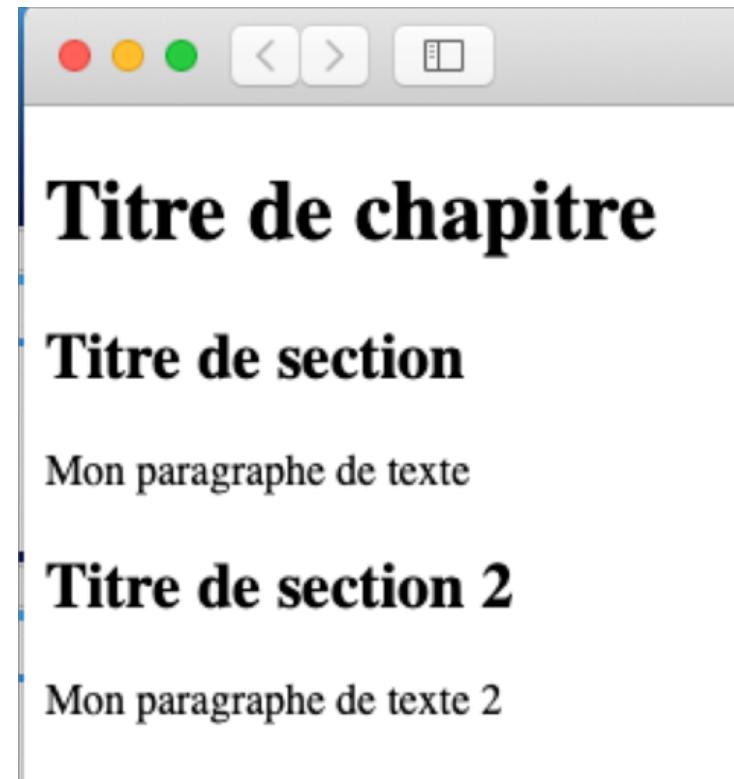
```
<h6>...</h6>
```

# HTML - Elements

## + Elements de texte

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
  </head>

  <body>
    <h1>Titre de chapitre</h1>
    <h2>Titre de section</h2>
    <p>Mon paragraphe de texte</p>
    <h2>Titre de section 2</h2>
    <p>Mon paragraphe de texte 2</p>
  </body>
</html>
```



# HTML - Elements

## + Elements de texte

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body>
    <h1>Titre de chapitre</h1>
    <h2>Titre de section</h2>
    <p>Mon paragraphe de texte</p>
    <h2>Titre de section 2</h2>
    <p>Mon paragraphe de texte 2</p>
  </body>
</html>
```

```
h1 {
  color: blue;
  background-color: yellow;
}
h2 {
  color: red;
}
p {
  color: #8080FF
}
```



# HTML - Elements

## + Les Listes

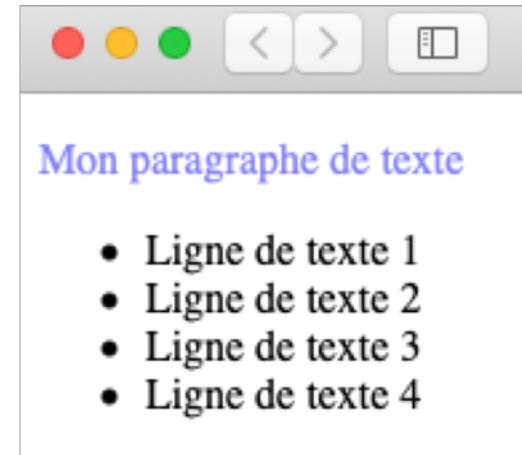
- + Définir une liste d'éléments
    - + Avec une puce : ul
    - + Liste numérotée : ol
  - + Une ligne dans la liste : li
- 
- + Définition de termes : dl
    - + Terme à définir : dt
    - + Définition : dd

# HTML - Elements

## + Elements de texte

```
<!DOCTYPE html>
<html lang="fr-fr">
    <head>
        <title>Test HTML</title>
        <link rel="stylesheet" type="text/css" href="Feuille.css" />
    </head>

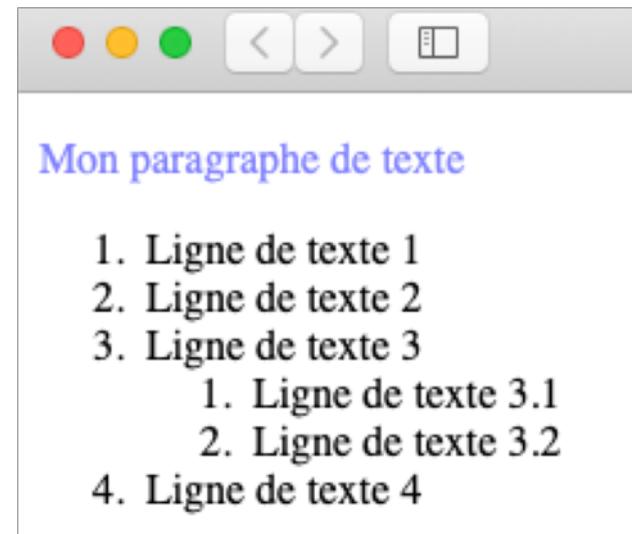
    <body>
        <p>Mon paragraphe de texte</p>
        <ul>
            <li>Ligne de texte 1</li>
            <li>Ligne de texte 2</li>
            <li>Ligne de texte 3</li>
            <li>Ligne de texte 4</li>
        </ul>
    </body>
</html>
```



# HTML - Elements

## + Elements de texte

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>
  <body>
    <p>Mon paragraphe de texte</p>
    <ol>
      <li>Ligne de texte 1</li>
      <li>Ligne de texte 2</li>
      <li>Ligne de texte 3
        <ol>
          <li>Ligne de texte 3.1</li>
          <li>Ligne de texte 3.2</li>
        </ol>
      </li>
      <li>Ligne de texte 4</li>
    </ol>
  </body>
</html>
```



# HTML

- + Mise en forme
  - + ID, classes, attribut style
  - + Retour à la ligne forcé : `<br />`
  - + Marquer un morceau de paragraphe : `<span>...</span>`
  - + Styles :
    - + Gras : `<b>... </b>`, `<strong>... </strong>`
    - + Italique : `<i>... </i>`, `<em>...</em>`
    - + Souligné : `<u>... </u>`
    - + ...
  - + Ligne de séparation : `<hr/>`

# HTML - Elements

## + Elements de texte

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

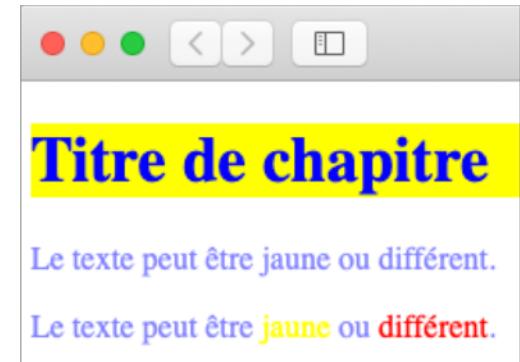
  <body>
    <h1>Titre de chapitre</h1>
    <p>Le texte peut être jaune ou différent.</p>
    <p>Le texte peut être <span style="color:yellow">jaune</span> ou
    <span class="erreur">différent</span>.</p>
  </body>
</html>
```

```
h1 {
  color: blue;
  background-color: yellow;
}

h2 {
  color: red;
}

p {
  color: #8080FF
}

.erreur {
  color : red;
}
```



# HTML - Elements

## + Images

### + Balise img

- + src = URL de l'image
- + alt = texte de remplacement si l'image n'est pas trouvée
- + Éventuellement height, width pour gérer la hauteur et la largeur

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body>
    <h1>Titre de chapitre</h1>
    <p></p>
  </body>
</html>
```



# HTML - Elements



img src='images/PoweredByMacOSX.gif' alt='Apple'/>



img src='images/PoweredByMacOSX.gif' alt='Apple' width="200"/>



img src='images/PoweredByMacOSX.gif' alt='Apple' height="200"/>



img src='images/PoweredByMacOSX.gif' alt='Apple' width="200" height="200"/>

# HTML - Elements

- + Les images utilisables :
  - + GIF : 8 bits compressés (max 256 couleurs)
  - + JPG, JPEG : 24 bits, compression par bloc de ressemblance  
Compression approchée
  - + PNG : 32 bits (1 octet / couleur + 1 octet pour la transparence)

# HTML- Elements

- + Lier les pages entre elles
  - + Définir des liens entre les pages html  
Cliquer sur un lien = demander la nouvelle page et l'afficher
- + Balise a
  - + href = l'URL du document
  - + Le contenu de la balise est l'élément sur lequel on peut cliquer
    - + Texte
    - + Image
    - + ...

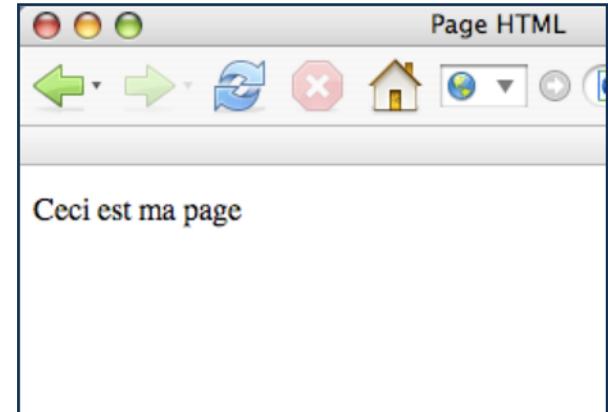
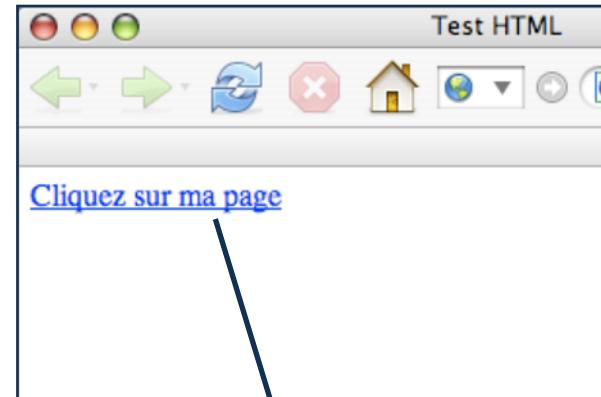
# HTML- Elements

## + Exemple

```
<!DOCTYPE html>
<html lang= "fr-frl">
  <head><title>Test HTML</title></head>
  <body>
    <p><a href="Page.html">Cliquez sur ma page</a></p>
  </body>
</html>
```

Page.html

```
<!DOCTYPE html>
<html lang= "fr-frl">
  <head><title>Page HTML</title></head>
  <body>
    <p>Ceci est ma page</p>
  </body>
</html>
```



# HTML- Elements

## + Exemple

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>
  <body>
    <h1>Titre de chapitre</h1>
    <p><a href="Page.html"></a></p>
  </body>
</html>
```

Page.html

```
<!DOCTYPE html >
<html lang= "fr-fr">
  <head><title>Page HTML</title></head>
  <body>
    <p>Ceci est ma page</p>
  </body>
</html>
```



# HTML- Elements

- + Faire apparaître le lien ailleurs que dans la fenêtre / l'onglet / la zone actuel
  - + Référence : <a href="page.html" target="**cible**">...</a>

Avec cible qui vaut

- + \_self = la zone actuelle (par défaut)
- + \_blank = dans une nouvelle fenêtre ou un nouvel onglet
- + \_parent = le contenant au dessus
- + \_top = la fenêtre / l'onglet qui contient la zone actuelle d'affichage
- + Tout autre nom de zone (défini dans un frame ou un iframe)

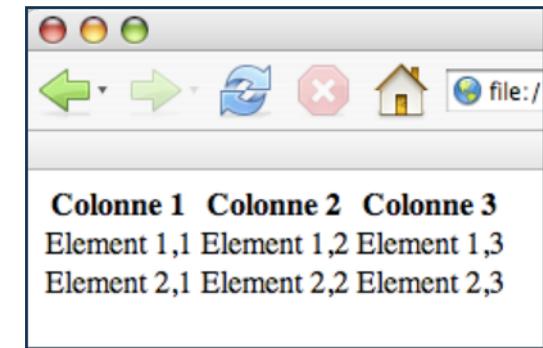
# HTML- Elements

- + Tables / tableaux
  - mise en œuvre de tableaux
    - + Entête de tableau <thead> ... </thead>
    - + Corps de tableau <tbody> ... </tbody>
    - + Fin de tableau <tfoot> ... </tfoot>
  - + Définir un tableau <table ...>
  - + Définir une ligne d'un tableau <tr...>
  - + Définir un entête de colonne de tableau <th ...>
  - + Définir une cellule <td ...>

<table>				
<tr>	<th> </th>	<th> </th>	<th> </th>	</tr>
<tr>	<td> </td>	<td> </td>	<td> </td>	</tr>
<tr>	<td> </td>	<td> </td>	<td> </td>	</tr>

# HTML- Elements

```
<!DOCTYPE html>
<html lang= "fr-frl">
  <head><title>Test HTML</title></head>
  <body>
    <table>
      <thead>
        <tr>
          <th>Colonne 1</th>
          <th>Colonne 2</th>
          <th>Colonne 3</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>Element 1,1</td>
          <td>Element 1,2</td>
          <td>Element 1,3</td>
        </tr>
        <tr>
          <td>Element 2,1</td>
          <td>Element 2,2</td>
          <td>Element 2,3</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```

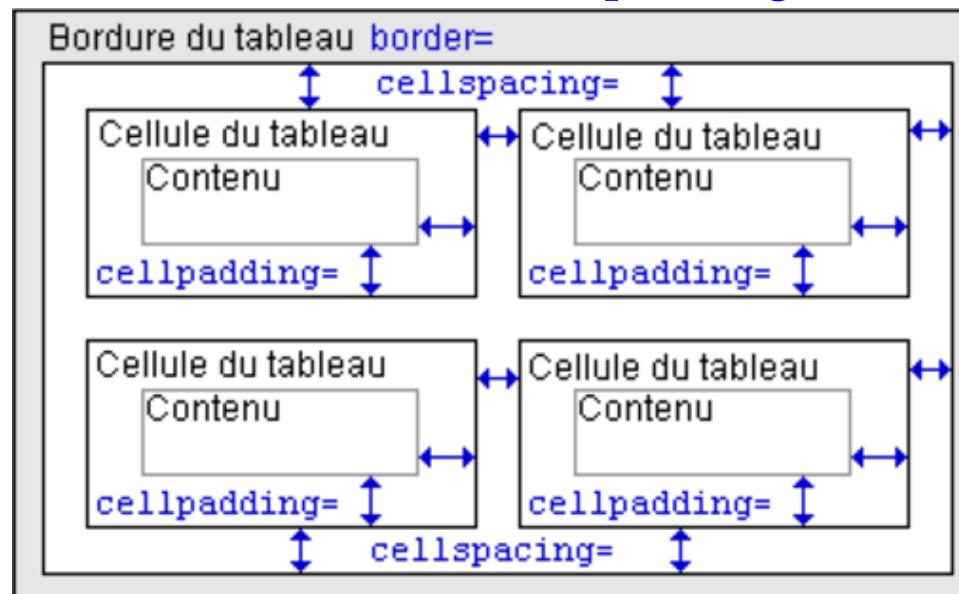


# HTML- Elements

## + Tables / tableaux

### Attributs de <table ...>

- + Distance entre cellules : `cellspacing="..."` (taille en points)
- + Espace autour de la cellule : `cellpadding= "..."` (taille en points)



# HTML- Elements

## + Tables / tableaux

Attributs de `<table border=".." ...>`

Gérer les bordures de tableau `frame="..."`

- + `void` : pas de bordure de tableau - intérieur uniquement.
- + `box` : tous les bords + intérieur
- + `above` : bord supérieur du tableau + intérieur
- + `below` : bord inférieur du tableau + intérieur
- + `hsides` : bords inférieur et supérieur du tableau + intérieur
- + `vsides` : bords droit et gauche du tableau + intérieur
- + `lhs` : bord gauche du tableau + intérieur
- + `rhs` : bord droit du tableau + intérieur

# HTML- Elements

## + Tables / tableaux

Attributs de <table border=".." ...>

Gérer le quadrillage de tableau rules="..."

- + **none** : pas de quadrillage de tableau.
- + **rows** : affichage des lignes entre les rangées
- + **cols** : affichage des colonnes
- + **all** : quadrillage complet
- + **groups** : entête, corps et fin de tableau  
necessite la présence des balises **thead**, **tbody** et **tfoot**

# HTML- Elements

## + Tables / tableaux

- + Fusion de lignes : <td rowspan="...">
- + Fusion de colonnes : <td colspan="...">

Attention : la fusion de ligne / colonne change le nombre de balises td dans une ligne

# HTML- Elements

```
<!DOCTYPE html>
<html lang= "fr-fr">
    <head><title>Test HTML</title></head>
    <body>
        <table border="1" style="background-color:yellow">
            <tr>
                <th style="background-color:green" colspan="2">Colonne 1</th>
                <th>Colonne 3</th>
            </tr>
            <tr style="background-color:red">
                <td>Element 1,1</td>
                <td>Element 1,2</td>
                <td rowspan="2">Element 1,3</td>
            </tr>
            <tr>
                <td>Element 2,1</td>
                <td style="background-color:blue">Element 2,2</td>
            </tr>
        </table>
    </body>
</html>
```

Colonne 1		Colonne 3
Element 1,1	Element 1,2	Element 1,3
Element 2,1	Element 2,2	

# HTML – Blocs et positionnement



# HTML - Blocs

## + Les Blocs

### + Balise div

Permet de regrouper des éléments pour leur appliquer des propriétés

### + Balises nav, section, aside, header, footer, ...

Dérivées des balises div.

## HTML - Blocs

### + Flux et positions relatives

Un certain nombre d'éléments forment des blocs en terme d'affichage

+ Les balises div

+ Les balises de titre h1.. h6

+ La balise p

+ Les balises de liste ul, ol, dl, li, dd, dt

+ Et quelques autres balises (address, pre, blockquote, ...)

+ Span, img, a, ...

bloc

En ligne

# HTML

- + Flux et positions relatives

- + **Balise bloc**

- + Positionnées habituellement les unes en dessous des autres
    - + Peuvent contenir d'autres balises et donc définir une hiérarchies de balises (père, enfant)

- + **Balises en-ligne**

- + Placent les éléments les uns à coté des autres

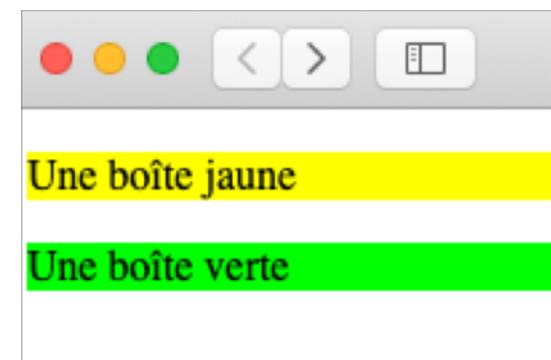
# HTML

## + Flux et positions relatives des blocs

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body>
    <p class="jaune">Une boîte jaune</p>
    <p class="verte">Une boîte verte</p>
  </body>
</html>
```

```
.jaune {
  background-color: #ffff00;
}
.verte {
  background-color: #00ff00;
}
```



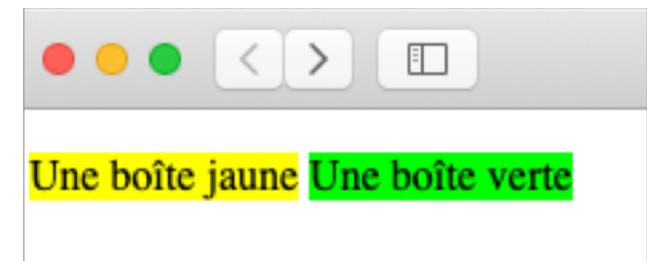
# HTML

## + Flux et positions relatives des éléments en ligne

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body>
    <p><span class="jaune">Une boîte jaune</span>
    <span class="verte">Une boîte verte</span></p>
  </body>
</html>
```

```
.jaune {
  background-color: #ffff00;
}
.verte {
  background-color: #00ff00;
}
```



# HTML

- + Flux et positions relatives
  - + Chaque élément d'une page peut être positionné
  - + Positionnement
    - + Positionnement dans le flux (défaut)
    - + Suivant un schéma de positionnement
      - + Absolu
      - + Flottant
      - + Relatif

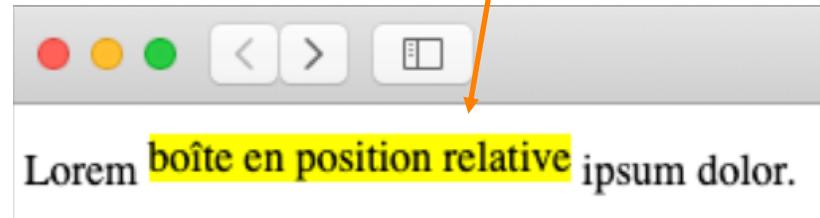
# HTML

## + Exemple :

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head>
    <title>Test HTML</title>
    <link rel="stylesheet" type="text/css" href="Feuille.css" />
  </head>

  <body>
    <p>Lorem
      <span class="jaune">boîte en position relative</span>
      ipsum dolor.</p>
  </body>
</html>
```

```
.jaune {
  position: relative;
  bottom: 5px;
  background-color: #ffff00;
}
```



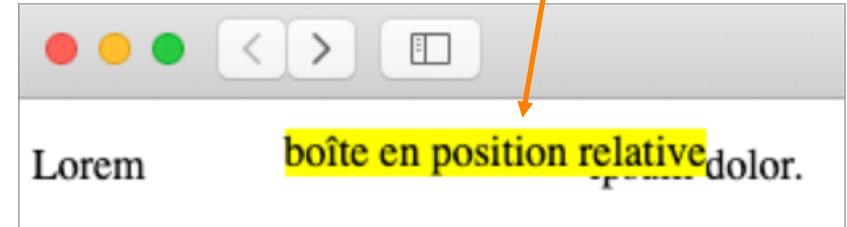
# HTML

## + Exemple :

```
<!DOCTYPE html>
<html lang="fr-fr">
    <head>
        <title>Test HTML</title>
        <link rel="stylesheet" type="text/css" href="Feuille.css" />
    </head>

    <body>
        <p>Lorem
            <span class="jaune">boîte en position relative</span>
            ipsum dolor.</p>
    </body>
</html>
```

```
.jaune {
    position: relative;
    bottom: 5px;
    left: 3em;
    background-color: #ffff00;
}
```



# HTML

- + Notion de positionnement

- + Positionnement **relatif**

Permet de positionner un élément **par rapport à la position qu'il aurait dû avoir** s'il avait été affiché sans positionnement.

**Position:relative**

permet de définir top, left, z-index

- + Positionnement **absolu**

Un bloc est positionné **par rapport au dernier ancêtre positionné** dans la hiérarchie.

**Position:absolute**

permet de gérer les dimensions (top, left, right, bottom, width, height, z-index, ...)

# HTML

- + Notion de positionnement

- + Position **fixe**

positionnement absolu **par rapport à la fenêtre**

Position:fixed

Permet de définir les attributs top, left

- + Position **statique**

Repositionnement d'un élément **par rapport à sa place normale**

Position:static

- + **Habillage**

Permet de mettre en oeuvre de l'habillage de texte

Float

Clear

# HTML

## + Notion de positionnement

- + Float : left

- Float : right

Permet de placer (faire flotter) l'élément à droite ou à gauche.

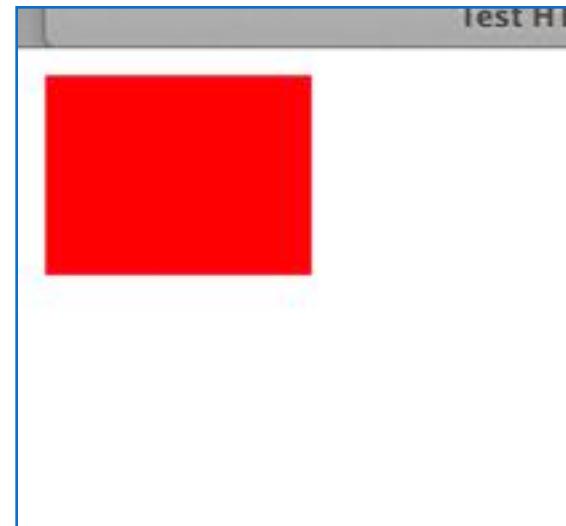
Attention : ceci est appliqué à partir de l'élément en question

=> pour tout remettre en état, on utilise Clear:both

# HTML

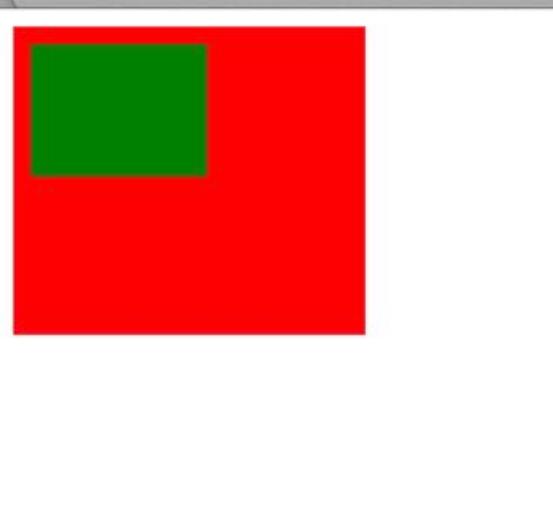
## + Position:absolute

```
<!DOCTYPE html>
<html lang="fr-fr">
  <head><title>Test HTML</title></head>
  <body>
    <div style="position: absolute;
                top: 10px; left: 10px;
                width: 100px; height: 75px;
                background-color: red;">
    </div>
  </body>
</html>
```



# HTML

## + Position:absolute

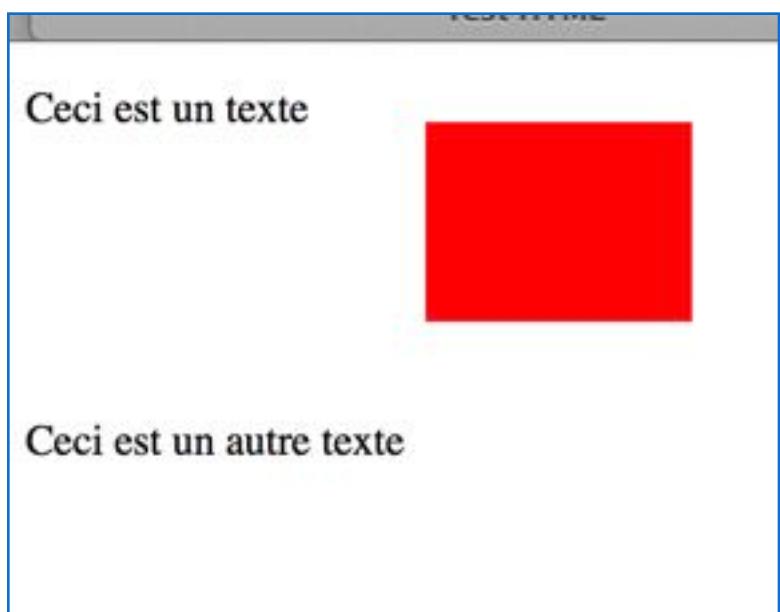
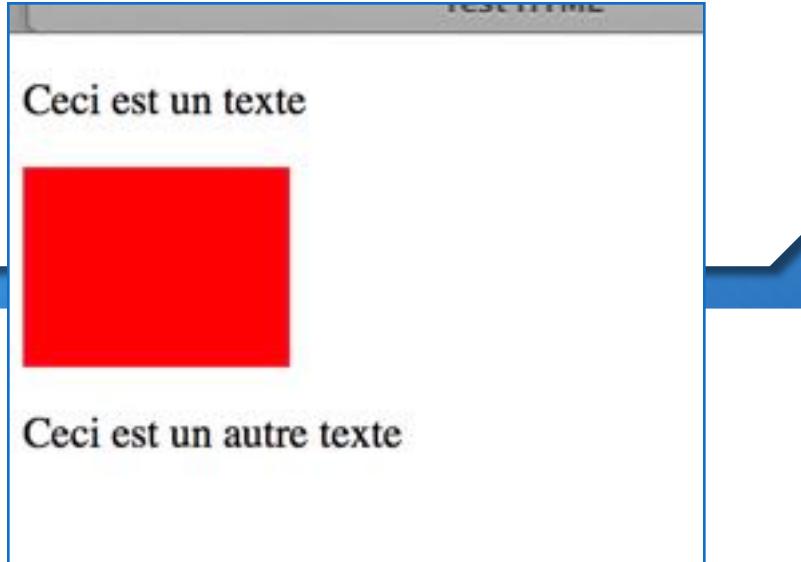


```
<!DOCTYPE html>
<html lang= "fr-fr">
    <head><title>Test HTML</title></head>
    <body>
        <div style="position:absolute;top:10px;left:10px; width:200px;height:175px;background-color:red;">
            <div style="position:absolute;top:10px;left:10px; width:100px;height:75px;background-color:green;">
                </div>
            </div>
        </body>
    </html>
```

# HTML

## + Position:relative

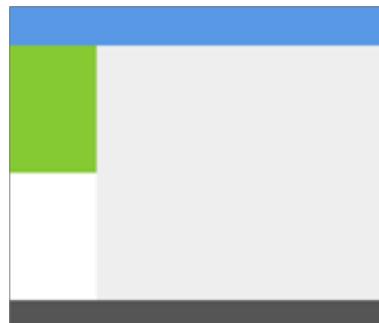
```
<!DOCTYPE html>
<html lang="fr-fr">
  <head><title>Test HTML</title></head>
  <body>
    <p>Ceci est un texte</p>
    <div style="position: relative;
      top:-20px;left:150px;
      width:100px;height:75px;
      background-color:red;">
    </div>
    <p>Ceci est un autre texte</p>
  </body>
</html>
```



# HTML

```
<div id="conteneur">
  <div id="header">
    <!-- Ceci est mon haut de page -->
  </div>
  <div id="sidebar">
    <!-- Ceci est ma colonne latérale -->
  </div>
  <div id="contenu">
    <!-- Ceci est mon contenu principal -->
  </div>
  <div id="footer">
    <!-- Ceci est mon pied de page -->
  </div>
</div>
```

```
#conteneur {width:760px; margin:0 auto;}
#sidebar {position:absolute; width:170px;}
#contenu {margin-left:170px;}
```



# Autres aspects

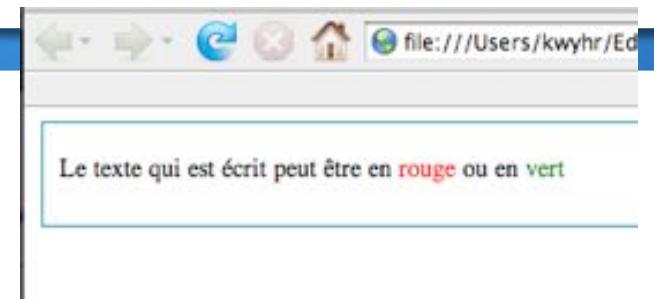


# HTML

## + Entourer une zone : fieldset

```
<!DOCTYPE html>
<html lang= "fr-frl">
    <head>
        <title>Test HTML</title>
    </head>

    <body>
        <fieldset style="border-style:double; border-width:1px; border-color:#0489a9;">
            <p>Le texte qui est écrit peut être en
            <span style="color:red">rouge</span>
            ou en <span style="color:green">vert</span>
            </p>
        </fieldset>
    </body>
</html>
```

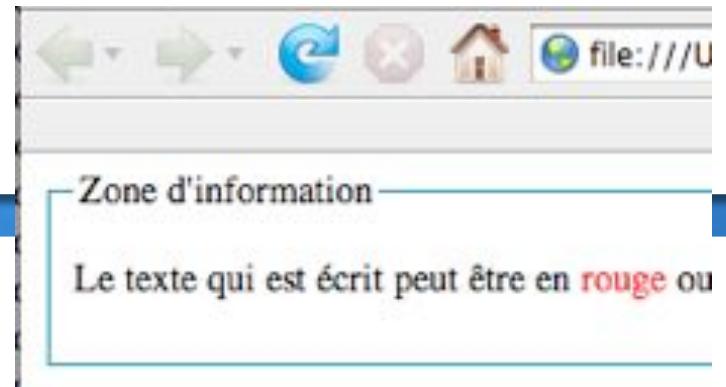


# HTML

## + Entourer une zone : fieldset

```
<!DOCTYPE html>
<html lang= "fr-frl">
    <head>
        <title>Test HTML</title>
    </head>

    <body>
        <fieldset style="border-style:double; border-width:1px; border-color:#0489a9;">
            <legend>Zone d'information</legend>
            <p>Le texte qui est écrit peut être en
                <span style="color:red">rouge</span>
                ou en <span style="color:green">vert</span>
            </p>
        </fieldset>
    </body>
</html>
```



# HTML

## + iFrame

Les iFrames permettent de définir une zone où sera affiché une page du même site ou d'un autre site.

# HTML

## + iFrame

Définition : <iframe ...>

- + Page source : `src="..."`
- + **Nom de la zone** : `name="..." / id = "..."`
- + hauteur : `height="..."`
- + largeur : `width="..."`
- + Bordure : `frameborder = "..."`
- + Ascenseur : `scrolling="..." YES | NO | AUTO`

# HTML

The screenshot shows a web page with a light blue header and a white main content area. In the top left corner of the content area, there is a small green decorative icon of a wreath. The main title 'Mes Recettes' is centered at the top in a large green font. Below the title, a green horizontal bar contains the text 'Recettes tirées de marmiton.org'. On the left side, there is a sidebar with three sections: 'Entrées' (with links to 'Bonhomme de neige' and 'Saint Jacques sur lit de poireaux'), 'Plats' (with links to 'Boeuf marengo' and 'Filets de truite à l'ananas et au basilic'), and 'Desserts' (with links to 'Tarte aux pommes, miel et amandes' and 'Gâteau léger au chocolat'). At the bottom of the sidebar, there is a black footer bar with the text '©Jean-Yves MARTIN - Ecole Centrale de Nantes - France'.

iframe

Le clic sur un lien affiche une nouvelle page dans l'iframe

# HTML

## + Afficher le contenu d'une page dans un iFrame : target

```
<!DOCTYPE html>
<html lang="fr-fr">
    <head>
        <meta http-equiv="content-type" content="text/html; charset=ISO-8859-1" />
        <link href="testiFrameStyle.css" rel="stylesheet" type="text/css" />
        <title>test IFrame</title>
    </head>
    <body>
        <div id="menu">
            <dl>
                <dd><a href="Page1.html" target="testContenu">Page 1</a></dd>
                <dd><a href="Page2.html" target="testContenu">Page 2</a></dd>
            </dl>
        </div>
        </form>

        <div id="contenu">
            <iFrame name="testContenu" id="testContenu" src="Page1.html">
            </iFrame>
        </div>
    </body>
</html>
```



# HTML - Les Formulaires



# HTML

## + Saisir des données ?

Comment faire pour

- + Saisir du texte (un nom, une adresse email, ...)
- + Choisir des éléments dans une liste ...
- + ...

=> Formulaires

# HTML

## + Formulaires

- + Permettent de placer des éléments permettant de saisir des données
  - + bouton
  - + menu déroulant
  - + zones de texte
  - + ...
- + Lorsque le formulaire est validé, les différentes informations collectées dans les éléments sont envoyées au serveur avec des instructions pour les traitements.
- + Le serveur exécute l'action prévue et retourne une réponse au navigateur.

# HTML

## + Déclaration d' un Formulaire

`<form ...>`

...

`</form>`

## Attributs :

- + **action** : nom de l' action à lancer sur le serveur
  - + Programme sur le serveur
  - + mailto...
- + **method** : mécanisme de passage des arguments
  - + =get : via l' URL
  - + =post : via la requête

# HTML

```
<!DOCTYPE html>
<html lang= "fr-frl">
    <head>
        <title>Test HTML</title>
    </head>
    <body>
        <form action="Action.php" method="post">
            ...
        </form>
    </body>
</html>
```

# HTML

## + Fonctionnement :

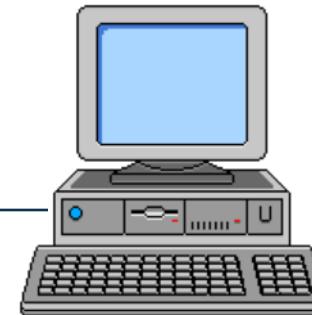
- + L'utilisateur saisit les valeurs des différents éléments du formulaire (textes, ...)
- + Lorsqu'il a fini, il valide le formulaire au moyen d'un des éléments prévu à cet effet
- + **TOUTES** les données saisies dans le formulaire sont envoyées au serveur
- + **SEULES** les données dans le formulaire sont envoyées au serveur
- + Le serveur effectue le traitement prévu et s'il y a lieu, retourne une nouvelle page html

# HTML

Client



Serveur



**Mon formulaire**

Nom

Prénom

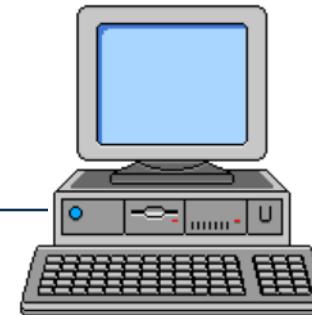
Email

# HTML

Client



Serveur



Mon formulaire

Nom	<input type="text" value="MARTIN"/>
Prénom	<input type="text" value="Jean-Yves"/>
Email	<input type="text" value="Jean-Yves.Martin@ec-nantes.fr"/>

Je remplis les données

Je clique sur le bouton Envoyer

# HTML

Client

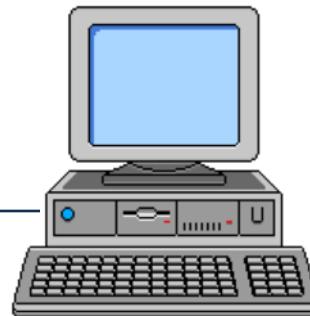


Mon formulaire

Nom	<input type="text" value="MARTIN"/>
Prénom	<input type="text" value="Jean-Yves"/>
Email	<input type="text" value="Jean-Yves.Martin@ec-nantes.fr"/>

```
<form action="Action.php" method="post">  
...  
</form>
```

Serveur



nom=MARTIN  
prenom=Jean-Yves  
email=Jean-Yves.Martin.ec-nantes.fr  
bouton=Envoyer

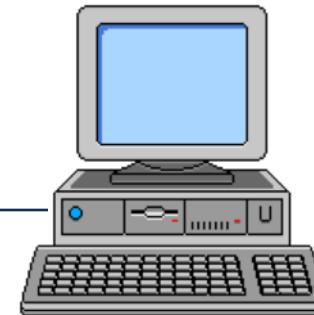
action=Action.php  
method=post

# HTML

Client



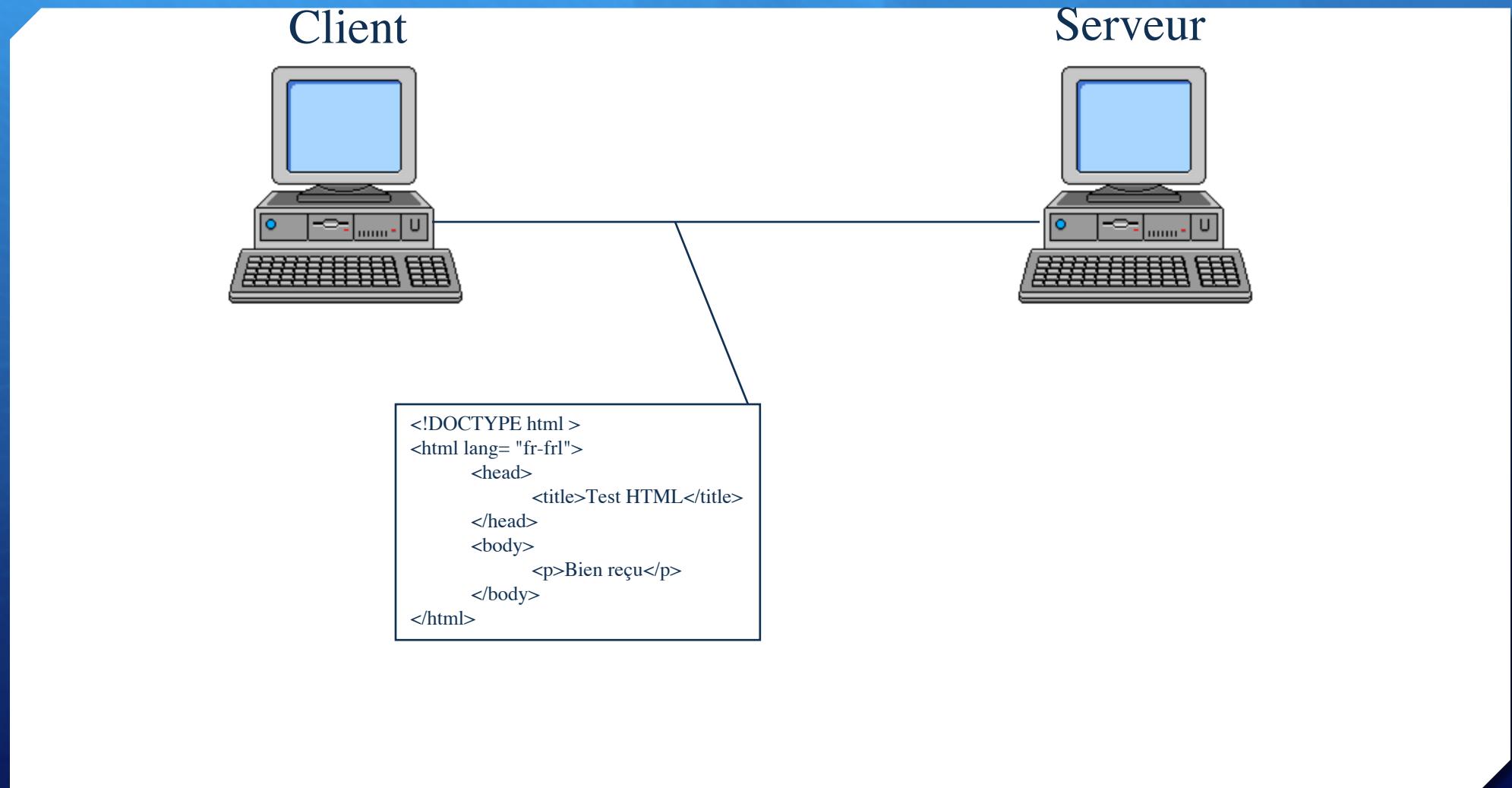
Serveur



```
nom=MARTIN  
prenom=Jean-Yves  
email=Jean-Yves.Martin.ec-nantes.fr  
bouton=Envoyer
```

Exécuter Action.php

# HTML

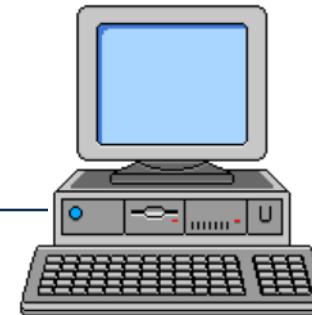


# HTML

Client



Serveur



Bien reçu

# HTML

## + Les éléments d'un formulaire `input`

### + Zones de saisie de texte

```
<input type="text" name="..." value="..." size="..." placeholder="..." />
```

Zone de texte

### + Mot de passe :

```
<input type="password" name="..." value="..." size="..." />
```

Password

### + Champ caché :

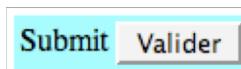
```
<input type="hidden" name="..." value="..." />
```

# HTML

- + Les éléments d'un formulaire  
**input**

- + Bouton de validation

```
<input type= »submit" name="..." value="..." />
```



- + Bouton de réinitialisation :

```
<input type="reset" name="..." value="..." />
```



- +

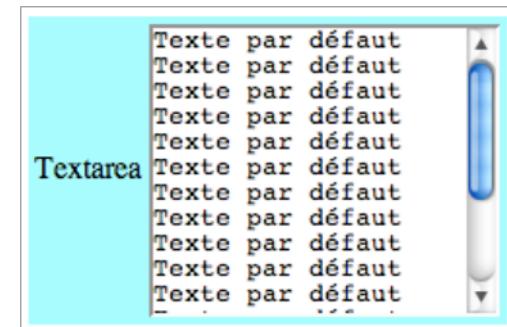
# HTML

## + Les éléments d'un formulaire

### + Zones de saisie de texte

#### + `textarea` :

```
<textarea name=".." rows=".." cols="..>  
...  
</textarea >
```



# HTML

## + Les éléments d'un formulaire

### + Bouton de radio :

```
<input type="radio" name="..." value="..." />
```

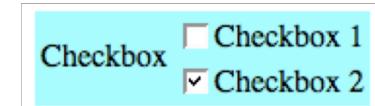
```
<input type="radio" name="..." value="..." checked="checked" />
```



### + Boîte à cocher :

```
<input type="checkbox" name="..." value="..." />
```

```
<input type="checkbox" name="..." value="..." checked="checked"/>
```

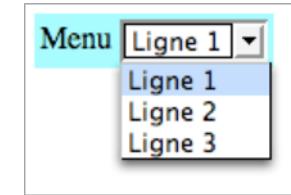


# HTML

## + Les éléments d'un formulaire

### + Menu déroulants :

```
<select name="...">
    <option value="...">>...</option>
    <option value="..." checked="checked">...</option>
    ...
</select>
```



# HTML

## + Traitement d'un formulaire

### + Clic sur un bouton RESET

Remet les valeurs de départ du formulaire

Ne provoque pas d'appel au serveur

### + Clic sur un bouton SUBMIT actif

Envoi une requête au serveur avec le nom de l'action à exécuter et la liste des valeurs des éléments du formulaire (name=value)

### + La touche RETURN active également l'envoi du formulaire

