



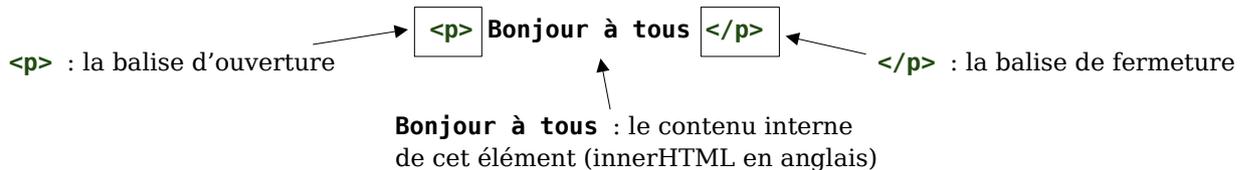
I – HTML : généralités

Le langage HTML est un **langage de description** : il décrit ce qu'on veut afficher.

Il est basé sur l'encapsulation des données dans des *balises ou éléments*.

Il existe deux types de balises :

→ Les **balises classiques** sont basées sur le fait d'avoir 3 parties :



DOCUMENTS 1 : Quelques balises (de type block)

<code><p> </p></code>	Paragraphes
<code><h1> </h1></code>	Titre principal
<code><h6> </h6></code>	Plus petit des sous-titres
<code> </code>	Liste ordonnée (qui devra contenir des lignes)
<code> </code>	Liste non ordonnée (qui devra contenir des lignes)
<code> </code>	Ligne

→ Les **balises orphelines** ne possèdent que la partie ouverture et n'ont ni contenu interne ni fermeture.

DOCUMENTS 2 : Quelques balises orphelines

<code>
</code>	Provoque un petit passage à la ligne (plus petit que p)
<code><hr></code>	Barre de séparation
<code></code>	Balise permettant d'insérer une image

→ Les **balises d'ouverture** peuvent contenir des informations supplémentaires sous la forme d'attributs :

DOCUMENTS 3 : Balises avec attributs

```

<a href="url_de_destination" target="_blank">Texte du lien</a>
<p style="background-color:yellow;">Des phrases</p>
```

II – HTML : les 3 catégories d'affichage

On notera également qu'il a 3 catégories de balises en terme d'affichage :

→ les balises **block** : provoquent un passage à la ligne et on peut régler **width** et **height**.

```
<p></p> <h1></h1> <li></li> <div></div> ...
```

→ les balises **inline** : ne provoquent pas de passage à la ligne et on ne peut pas régler **width** et **height**.

```
<a></a> <span></span> <strong></strong> <em></em> <mark></mark> ...
```

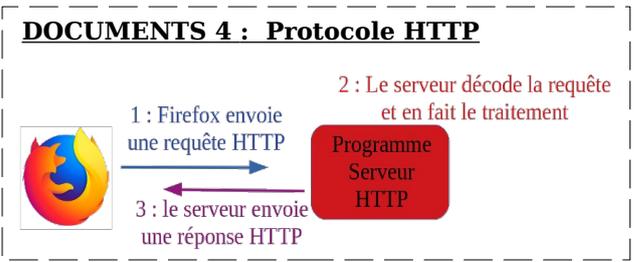
→ les balises **inline-block** : ne provoquent pas de passage à la ligne mais on peut régler **width** et **height**.

```
<img>
```

III – Principe du HTTP

- 0 - Le serveur écoute.
- 1 - Le client lance une **REQUETE HTTP** au serveur.
- 2 - Le serveur répond avec une **REPONSE HTTP** contenant éventuellement un **code-source HTML**.
- 3 - Le client reçoit et analyse cette réponse.

Si le code-source HTML contient des références à des images, des fichiers CSS ou autres, le navigateur va alors lancer d'autres requêtes pour obtenir les requêtes manquantes.



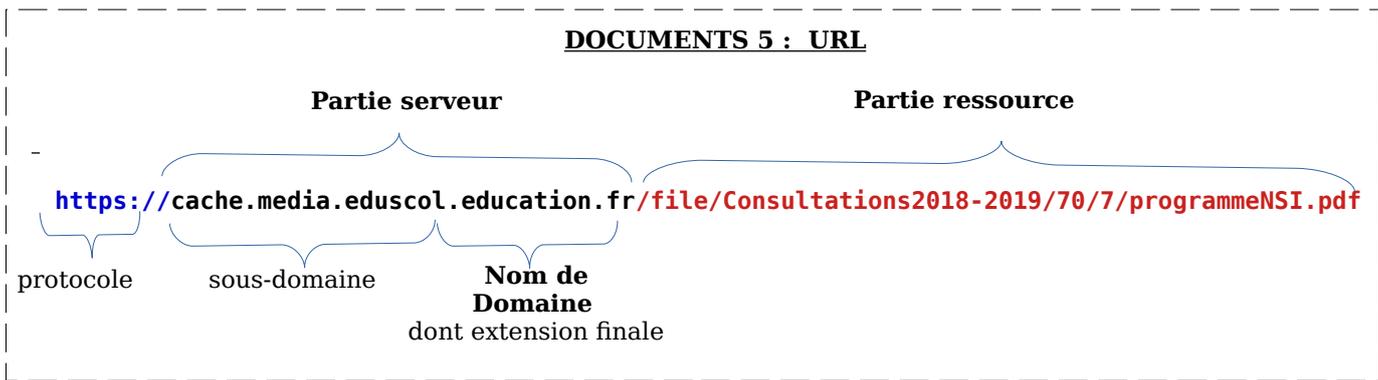
IV – URL : Uniform Ressource Locator

Trois façons de localiser une ressource :

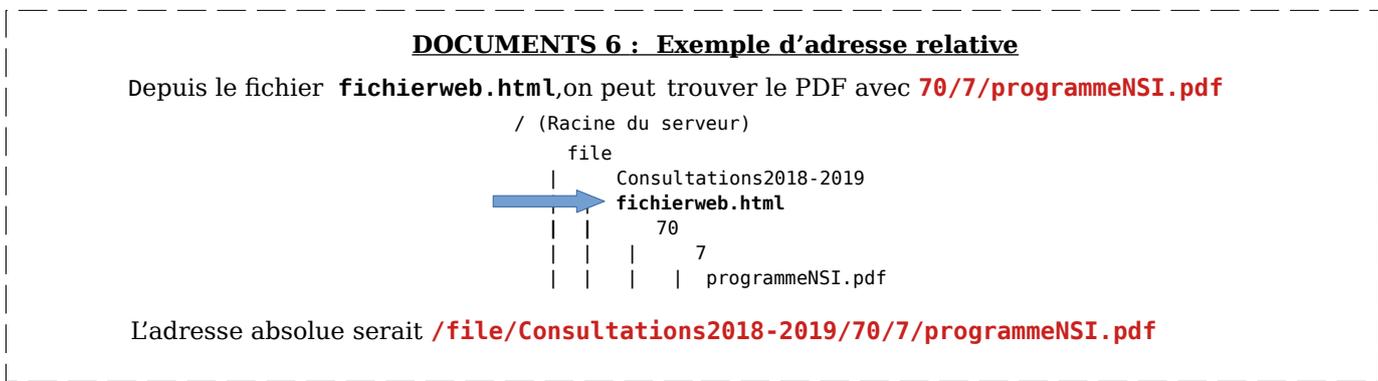
- en fournissant l'**URL complète** : localisation sans ambiguïté sur Internet.
- en fournissant **une adresse absolue** : localisation depuis **la racine du serveur** .
- en fournissant **une adresse relative** : localisation depuis **par rapport** à la localisation du fichier HTML actuel.

URL complète : composée du **protocole**, de l'**adresse du serveur**, et de l'**adresse absolue de la ressource** demandée sur ce serveur.

Si l'adresse ressource commence par **un slash /**, on dit qu'il s'agit d'une **adresse ABSOLUE** par rapport à la racine du serveur.



Adresse relative : On fournit l'adresse de la ressource par rapport à la localisation actuelle de la ressource HTML initiale. Cette adresse **ne commence donc pas par un slash**.



V – Web Sémantique : donner du sens aux mots et à leur position dans la page

HTML intègre des balises permettant d'aider les robots des moteurs de recherche à comprendre de quoi parlent vraiment les pages.

Il existe ainsi des **balises de type block** qui ne possèdent pas d'affichage propre mais donne un sens au contenu interne qu'il stocke.

<head>	→ contient des informations qu'on veut transmettre au navigateur, sans affichage direct
<body>	→ contient toutes les balises qui doivent provoquer un affichage
<header>	→ l'en-tête de la page, signale une introduction.
<nav>	→ contient des liens de navigation
<main>	→ contient le corps de la page, ce dont la page parle vraiment
<article>	→ contient une partie qui peut se lire de façon indépendante normalement
<section>	→ contient une partie qui ne peut pas se lire de façon indépendante normalement
<aside>	→ contient une partie annexe au contenu réel, un apparté.
<footer>	→ contient le pied de page, la conclusion

Il existe également des balises sémantiques de type inline (qui possèdent elles un affichage par défaut différent du texte standard) :

	→ emphase / assez important. Par défaut en italique.
	→ important. Par défaut en gras.
<mark>	→ très important. Par défaut en surligné jaune fluo.

VI – CSS : Cascading Style Sheet

Le HTML précise le FOND, le contenu de la page.

Le CSS précise la FORME, l'apparence que doit avoir ce contenu.

On peut inclure du style CSS de 3 façons :

- directement dans une balise en utilisant l'attribut **style**. A éviter et à réserver au prototypage.
- dans une balise **<style>** intégrée à la page HTML. A éviter car le style ne s'appliquera qu'à cette page.
- **dans un fichier CSS** localisé à l'aide d'une balise **<link>** située dans la balise **<head>** de la page HTML.

DOCUMENTS 7 : Comment préciser l'adresse du fichier CSS

```
<!DOCTYPE html>
  <html lang="fr">
    <head>
      <meta charset="UTF-8" />
      <title>CSS à part (page 2)</title>
      <link rel="stylesheet" href="style_1.css" />
    </head>
    <body>
      ...
    </body>
  </html>
```

On précise le type des balises sur lesquelles on veut agir et on place entre accolades les styles CSS qu'on veut leur imposer. Exemple :

header, footer	→ Toutes les balises header et footer
{	
background-color : #AAAA00;	→ auront un fond coloré jaune (rouge et vert à fond FF)
color : #5500AA;	→ et une écriture de texte plutôt bleu-violet
}	

Pour vous renseigner sur des effets style CSS, une recherche « w3school + effet voulu ».

VII – CLASSE : Différenciation d'effets CSS sur des balises de même type

Si vous ne souhaitez pas que toutes les balises p aient le même style CSS, il faut les différencier en utilisant un attribut HTML nommé **class**. On pourra alors faire agir le CSS sur les balises ayant cette classe.

DOCUMENTS 8 : Exemple d'actions du CSS

```
HTML      <p>Paragrahe1.</p>
          <p class="typeA">Paragraphe 2.</p>
          <p class="typeA">Paragraphe 3.</p>

CSS       .typeA      - on notera la présence du point pour signaler CLASS typeA et pas balise typeA
          {
            font-style: italic;
            background-color: grey; color : white;
          }

          header, footer
          {
            background-color: #AAAA00;
            color: #5500AA;
          }
```

VIII – Balises génériques

Pour agir sur un ensemble de balises, on peut les englober dans **des balises-conteneurs génériques** et agir sur le style CSS de la balise conteneur elle-même : **les enfants héritent des propriétés CSS** du parent.

La balise générique <div> est une balise block.

La balise générique est une balise inline.

VIII – Compléments CSS

DOCUMENTS 9 : Quelques propriétés CSS très communes

Colueur écriture	color: #AAAA00;
Fond coloré	background-color: #AAAA00;
Type écriture	font-style: italic normal;
Soulignement...	text-decoration: none underline overline line-through;
Taille écriture	font-size: xx-small x-small small medium large x-large xx-large ;
Gras ou pas	font-weight: bold normal;
Blocks	
Centrage	text-align: left right center justify;
Ecartement des autres balises	margin: 20px; et margin-top, margin-left...
Espace interne supplémentaire	padding: 20px; et padding-top, padding-left...