

CONSTRUCTION DE SITES WEB

6ème partie

Sommaire

Le JavaScripts	4
Notions de Base JavaScript	4
1. Comment ça marche ?	4
2. Avantages et inconvénients	4
3. Intégrer du Javascript dans une Page HTML.....	5
Introduction aux Feuilles de Style	8
Intégrer une Feuille de Style	9
Appliquer une Feuille de Style	10
Les Images Mapées	14
1. Les Balises MAP	14
2. La Zone MAP	14



Jean-Pierre Pons - Janvier 2005



ORT Mondiale
1, rue de Varembe - 1211 Genève 20 Suisse
Tél : 41 22 - 919 42 34
Fax : 41 22 - 919 42 39
Email : ortcoop@bluewin.ch

LE JAVASCRIPT

C'est quoi un Javascript ?

Javascript a été créé par Netscape (le nom était livescript au départ) pour développer des applications Internet et, par la même occasion, pour étendre les possibilités du HTML. Il s'agit d'un langage léger et orienté objet, permettant d'écrire des scripts. Attention : il ne faut pas confondre Java et Javascript ! Java est un langage de programmation développé par SUN et non un langage de script comme Javascript. De plus, les scripts écrits avec Javascript sont interprétés par le navigateur et incorporés dans le code HTML, ce qui n'est pas le cas du langage Java.

Aujourd'hui le javascript repose sur un standard: ECMAScript. Pour cette raison je parlerais sur ce site soit de javascript soit de ECMAScript pour parler sur langage de script utilisé pour inclure des programme dans les pages html. Une fois que vous aurez lu ces pages je vous conseille de vous pencher sur le standard: standard ECMAScript

Dans cette section, je vous propose d'apprendre à utiliser ce langage, cela s'adresse donc plus spécialement aux personnes ayant quelques bases en programmation.

Après la lecture de ces pages vous serez en mesure d'écrire de petits programmes en javascript, si vous souhaitez interagir avec les éléments constituant votre page html, rendez vous dans la section DHTML.

Note: les exemples utilisent la commande `document.write(...)` qui permet d'afficher du texte dans la page html

Comment intégrer un programme en javascript dans une page html

Il existe deux façons d'inclure un programme en javascript dans votre page html. La première consiste à écrire directement votre programme dans la page :

```
<script type="text/javascript">  
// ici votre programme en javascript  
</script>
```

La deuxième consiste à écrire votre programme dans un fichier indépendant (script.js dans l'exemple ci-dessous) et à demander au navigateur de le télécharger:

```
<script type="text/javascript" src="script.js"></script>
```

dans ce dernier cas on place généralement la directive dans la section head de la page html. L'avantage de cette méthode est que le fichier ne sera chargé qu'une seule fois par le navigateur et stocké dans son cache si il apparait dans plusieurs pages.

NOTIONS DE BASE JAVASCRIPT

1. COMMENT ÇA MARCHE ?

Javascript est un langage qui vient compléter le HTML avec de l'interactivité, il n'est pas compilé mais juste interprété. C'est un langage purement événementiel, chaque Javascript étant toujours déclenché par un événement (clic sur un bouton, sur un lien, chargement d'une page, etc...).

A l'instar du HTML et contrairement au PHP ou au Perl, le langage Javascript est interprété au niveau du client, c'est à dire le navigateur Web.

Durant ce cours, vous aurez souvent à faire avec des objets et des méthodes. Cela provient du fait que Javascript est un langage orienté objet. Que cela ne vous déroute pas, une méthode est tout simplement une fonction qui va agir sur un objet qui peut être une variable, un tableau ou tout élément d'une page HTML.

2 AVANTAGES ET INCONVÉNIENTS

Comme tous les langages, Javascript possède ses avantages et ses inconvénients, en voici un aperçu:

Avantages:

- c'est un langage facile à maîtriser,
- il ne surcharge pas le serveur sur lequel se trouve le site,
- il permet de rendre un site plus vivant et plus fonctionnel.

Inconvénients:

- son source est accessible à tous les visiteurs du site,
- certains navigateurs web ne supportent pas pleinement ou pas du tout ce langage
- il peut y avoir des incompatibilités entre les différents navigateurs et les différentes plateformes.

3. INTÉGRER DU JAVASCRIPT DANS UNE PAGE HTML

Il existe différentes manières d'intégrer du code Javascript dans vos pages HTML. Ces exemples vous aideront à comprendre chaque possibilité.

Balises `<script></script>`:

```
<SCRIPT type="text/javascript">
<!-- les instructions;javascript; ici; //-->
</SCRIPT>
```

`<NOSCRIPT>`

Ce qui s'affichera si le navigateur n'est pas compatible Javascript.

`</NOSCRIPT>`

Les balises `<SCRIPT></SCRIPT>` se placent en général (mais pas obligatoirement) entre les balises `<HEAD>` et `</HEAD>` et permettent d'indiquer au navigateur que la page contient un script, on précise le type de script avec la propriété `type="text/javascript"` de la balise ouvrante `<SCRIPT>`. Ces balises peuvent directement contenir des instructions ou des fonctions Javascript.

Il est conseillé de mettre les instructions à exécuter entre commentaires grâce à `<!--` (début du commentaire) et `-->` (fin du commentaire). Cela permet de ne pas afficher le code sur les navigateurs ne supportant pas ce langage.

Les balises `<NOSCRIPT></NOSCRIPT>` permettent d'afficher un texte quelconque toujours dans le cas où le navigateur ne supporterait pas le Javascript.

Insertion de code dans une URL d'un élément HTML

Le code se met sous la forme :

```
javascript:code
```

Élément A

Pour exécuter du code JavaScript il est possible d'utiliser :

```
<A href=»JavaScript:code»>texte</A>
```

Quand on clique sur texte le code s'exécute (le plus souvent c'est un appel de fonction).

INTRODUCTION AUX FEUILLES DE STYLE

Les feuilles de styles (en anglais CSS : Cascading Style Sheet) permettent de définir la représentation graphique d'un élément HTML. Vous contrôlez ainsi la représentation de votre document dans le navigateur de vos visiteurs (mais aussi la façon dont le document sera imprimé...)

Cette technologie est définie dans une norme élaborée par le W3C (World Wide Web Consortium). Il en existe plusieurs versions CSS1 pour la version 1, CSS2 (version 2) ...

Un petit exemple pour mieux comprendre :

la méthode classique pour écrire en bleu est la suivante :

```
<font color=»blue»>ceci est bleu</font>
```

ce qui donne :

ceci est bleu

A présent avec l'aide d'une feuille de style nous allons forcer l'écriture de toutes les sections DIV en bleu. Cela s'écrit de la façon suivante :

```
<style>
```

```
div.test { color: blue; }
```

```
</style>
```

```
<div class=»test»>ceci est bleu</div>
```

Le résultat est le même :

ceci est bleu

Petite explication : la section style permet de définir une feuille de styles. On spécifie ensuite le tag html et les attributs que l'on souhaite modifier (ici color). Et voilà le tour est joué. C'était votre première feuille de style.

INTÉGRER UNE FEUILLE DE STYLE

Dans la section précédente nous avons vu que nous pouvions spécifier une feuille de style à l'aide du tag `style`, ce qui donne :

```
<style>
```

```
div { color: blue; }
```

```
...
```

```
</style>
```

cette section se trouve théoriquement dans la section HEAD de votre document HTML, pour mémoire on obtient donc un document de cette forme

```
<html>
```

```
<head>
```

```
<title> le titre de ma page </title>
```

```
...
```

```
<style>
```

```
...
```

```
</style>
```

```
</head>
```

```
<body>
```

```
... mon document html ...
```

```
</body>
```

```
</html>
```

il existe une deuxième méthode, qui consiste à faire un lien vers un fichier qui contient toutes les informations de description de la feuille de style. Cette méthode est préférable car de cette façon le navigateur ne chargera la feuille de styles qu'une fois pour toute pour votre site. La syntaxe est la suivante :

```
<link href=»style.css» type=»text/css» rel=»stylesheet» >
```

ici le fichier utilisé est 'style.css'

Une dernière méthode consiste à spécifier le style directement comme un attribut du tag :

```
<h3 style=»color: red;»>texte en rouge format h3</h3>
```

APPLIQUER UNE FEUILLE DE STYLE

Voyons à présent comment écrire une feuille de style. Une feuille de styles s'appliquant à un document html, il faut d'abord être au point avec cette première syntaxe. Voyons un petit exemple d'html sur lequel sera basé notre feuille de styles.

```
<h3>coucou</h3>
<div class=»c1«>
jaune
<div id=»id1«>centr&eacute;</div>
<div id=»id2«>vert</div>
<div>bleu</div>
<h3>hello</h3>
</div>
<h3 class=»c2«>coucou c2</h3>
```

Quelques explications sont peut être nécessaires. Nous avons d'abord un titre au format h3. Ensuite nous avons une section de class c1 qui inclus trois sections dont une qui a l'ID id1. L'attribut class permet de définir une classe d'appartenance de l'objet, il sera généralement utilisé pour un ensemble d'objets. L'attribut ID permet de définir une feuille de style pour un objet bien précis, vous l'utiliserez ci vous n'avez qu'un ou deux éléments qui ont une représentation spécifique.

A cela s'ajoute la notion d'arborescence, dans l'exemple ci-dessus, on peut dire que id1 est l'attribut d'un objet situé sous un objet de classe c1. Dans une feuille de style on pourra faire la différence entre un objet d'id id1 selon qu'il sera situé dans une section c1 ou pas.

Assez de théorie, écrivons une feuille de style qui corresponde à notre document :

```
<style>
h3 { color: blue; text-decoration: underline; font-size: 10pt;}
div h3 { color: red; }
.c1 { color: yellow; background: green; font-weight: bolder;}
#id1 { text-align: center; }
.c1 #id2 { color: green; background: white; }
.c1 div { color: blue; }
h3.c2 { background: yellow; }
</style>
```

ce qui nous donne :



Cet exemple est très complet et propose toutes les combinaisons que vous pouvez avoir sur une feuille de styles. Commencez par utiliser la première déclaration toute simple, et ensuite quand vous serez bloqués, revenez vers l'exemple pour voir tout ce que vous propose les Cascading Style Sheet.

connaître toutes les possibilités

Le W3C a édité une norme concernant les CSS, vous pouvez donc sur leur site W3C retrouver toutes les spécifications concernant cette technologie.

Dans les exemples précédents nous avons vus des exemples utilisant color, text-decoration ... dans la norme vous retrouverez toutes les noms des propriétés et les valeurs possibles.

Il n'est pas toujours simple de voir quelle sera la représentation d'un élément un fois qu'une propriété lui aura été appliquée. N'hésitez pas à utiliser l'attribut style pour faire des tests rapides.

```
<h3 style="color: blue; font-size: 10pt;">
```

Exemple 1 : Titre

```
<style>
h3 { font-family: Verdana; font-size: 10pt; font-weight: bold; color: #D60000; margin: 0px; }
</style>
```

On choisit les caractéristiques de la police (font) en choisissant sa famille (Verdana) la taille (10) et le style (gras). Ensuite on définit la couleur avec les composantes rouge/vert/bleu et on choisit de supprimer la marge autour de l'objet h3.

Exemple 2 : Liens

```
<style>
a { font-family: Verdana; font-size: 8pt; color: #D60000; text-decoration: none; }
a:hover { color: #ffe7ce; text-decoration: underline; }
</style>
```

Dans cet exemple la couleur du texte correspondant au lien changera au moment où la souris sera au dessus et le texte sera souligné. On pourrait aussi changer tous les autres attributs d'un lien et changer le style pour qu'il devienne écrit en gras par exemple.

La propriété text-decoration supporte les valeurs suivantes :

- * none : rien
- * underline : souligné
- * overline : surligné
- * line-through : barré
- * blink : clignotant

Exemple 3 : scrollbar d'Internet Explorer

Internet Explorer offre une extension à la norme intéressante puisqu'elle permet de modifier l'apparence de la barre de défilement de votre navigateur. exemple:

```
<style>
body { scrollbar-arrow-color:d60000; scrollbar-track-color:daedf4; scrollbar-shadow-color:5aadce; scrollbar-face-color:8ec7dc; scrollbar-highlight-color:daedf4; scrollbar-darkshadow-color:215263; scrollbar-3dlight-color:daedf4; }
</style>
```

Vous n'avez plus qu'à modifier les couleurs pour votre site.

Exemple 4 : table

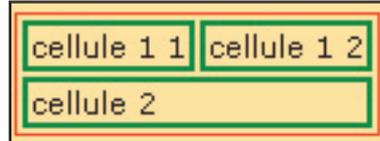
Les tables sont un sujet très intéressant pour les feuilles de styles car on peut gérer de façon très fine leur présentation.

Une table est composée de 3 tags: table, tr, td. On peut modifier la presentation de chacun

voici un exemple :

```
<style>
table { border-spacing: 0px; border-collapse: collapse; padding: 0px; border: 1px red solid; }
td { font-family: Verdana; font-size: 8pt; color: #2E2024; margin: 0px; padding: 0px; border: 2px green solid; }
</style>
```

ce qui donne :



cellule 1 1	cellule 1 2
cellule 2	

Exemple 5 : Ecran et Impression différents

Les feuilles de styles peuvent être utilisées pour obtenir un rendu de votre site différent selon la méthode utilisée pour le visualiser. Il est par exemple possible d'avoir un rendu différent quand on imprime une page et quand on le visualise dans un navigateur. La gestion de l'impression est sans doute la fonctionnalité la plus intéressante car il est même possible de gérer les sauts de page dans un document html.

Pour voir la différence on utilise l'attribut media, ce qui donne

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

ou

```
<link href="style.css" type="text/css" rel="stylesheet" media="print" >
```

screen correspond à la visualisation dans le navigateur et print à l'impression.

LES IMAGES MAPÉES

Les images mapées ou cliquables ont la particularité d'avoir des zones sur lesquelles il est possible de cliquer et d'ouvrir ainsi une page située à une adresse (URL) déterminée. Ce procédé s'appelle «hypergraphique» car il permet d'insérer un lien hypertexte dans une image.

Un petit exemple, prenons la carte d'Afrique du site de l'ORT : <http://ortcoop.free.fr>

En passant votre souris sur l'image, vous voyez que selon l'endroit où vous cliquez, vous n'arrivez pas sur la même page. Cette image est mapée car elle est découpée en zone comme une carte (Map en Anglais).

1. LES BALISES MAP

Il existe plusieurs méthodes pour réaliser cet effet d'images mapées : les méthodes NCSA, CERN et CSIM. La plus courante est la méthode CSIM (Client Side Image) car elle est définie dans le HTML 3.0.

La balise <MAP est divisée en deux :
d'une part l'image :

```
<IMG SRC=»image.gif» USEMAP=»#nom»>
```

d'autre part les zones

```
<MAP NAME=»nom»>  
<AREA SHAPE=»forme» COORDS=»X1,Y1,X2,Y2» HREF=»URL»>  
<AREA...  
</MAP>
```

Expliquons tout cela:

1. L'image

Pour l'image, c'est simple. On utilise le même code que pour une image normale (Cf : affichage d'une image), sauf que l'on rajoute l'attribut USEMAP qui indique le nom du MAP que vous allez utiliser. N'oubliez pas le «#».

2. LA ZONE MAP

Là, cela se complique. Tout d'abord, il y a la balise <MAP>...<MAP>. Dans cette balise, il faut indiquer un nom dans l'attribut NAME. IL faut que ce soit le même que celui donné dans le USEMAP de votre image.

Ensuite, il y a les balises <AREA>, une pour chaque zone sur votre image.

L'attribut SHAPE définit la forme de votre zone : rect (rectangle), circle (cercle), polygon (polygone irrégulier).

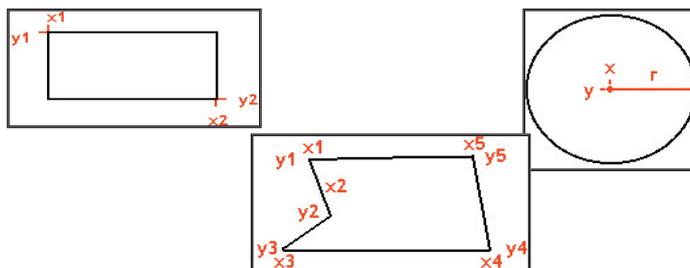
l'attribut HREF définit l'adresse vers laquelle va pointer votre zone.

l'attribut COORDS indique les coordonnées de la zone en pixels. Elles varient selon la forme géométrique que vous avez choisie.

pour le rectangle: COORDS=»x1,y1,x2,y2»

pour le cercle: COORDS=»x,y,r»

pour le polygone: COORDS=»x1,y1,x2,y2,...»



l'attribut ALT est facultatif et permet d'afficher un commentaire quand la souris s'arrête au-dessus de la zone (surtout pour les butineurs récents).

Le code Map peut être inséré n'importe où dans la page, mais le mieux est le mettre tout de suite derrière l'image. ainsi le code du logo précédent était:

```
<IMG SRC=»../logomain.gif» USEMAP=»#logomap» ALT=»logomain» BORDER=0 HEI-
GHT=65 WIDTH=66>
<MAP NAME=»logomap»>
<AREA SHAPE=»rect» COORDS=»0,0,38,30», HREF=»../starshoot.htm»
ALT=»starshoot»>
<AREA SHAPE=»rect» COORDS=»39,0,66,30», HREF=»../amis.htm» ALT=»friends»>
<AREA SHAPE=»rect» COORDS=»39,31,66,65», HREF=»../contacts.htm»
ALT=»contacts»>
<AREA SHAPE=»rect» COORDS=»0,31,38,65», HREF=»../technique.html»
ALT=»technique»>
</MAP>
```

Faites aussi attention de ne pas recouvrir un même endroit avec deux zones. Au mieux, il ne se passera rien ; au pire, il y aura un message d'erreur du navigateur.

Les Logiciels

Le code pour une image mapée n'est pas facile à réaliser ; il est donc intéressant d'avoir un petit logiciel pour le faire.

Ici aussi de petits logiciels comme MapMaker, Mapedit, MapThis, CrossEye, Web Hotspots... vous seront d'une grande utilité.

