

<https://vader-fr.fr/spip.php?article38>



II) Structure d'un fichier CSS

- Développement web - Les feuilles de style CSS -



Date de mise en ligne : dimanche 25 mars 2012

Copyright © Vader[FR] : ce n'est pas un blog, c'est un Sith - Tous droits

réservés

II) Structure d'un fichier CSS

Un fichier css contient les éléments, id, classes et évènements utilisés sur une page web, auxquels on associe des attributs de style et leur valeur.

- un bloc d'attributs de style associés à un élément est situé entre deux accolades "{ }" après la déclaration dudit élément.
- un attribut est séparé de sa valeur par deux points ":"
- les attributs sont séparés entre eux par des points-virgule ";"

On peut mettre tous les attributs sur une même ligne, mais il est plus propre de déplier le bloc et d'indenter le code.

ce qui donne :

```
élément {  
  attribut:valeur;  
}
```

Un élément quelconque, en tant que tel, sera indiqué par le nom de sa balise html ([...] donnera img, donnera a, donnera div, etc).

Par exemple :

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

mettra une couleur d'arrière-plan en noir pour l'élément body, qui correspond au corps de la page.

A noter, si le contenu ne descend pas jusqu'en bas de la fenêtre, il n'y aura qu'une partie de la fenêtre en arrière-plan noir. (Voir article sur la gestion de la taille des éléments web)

un identifiant d'élément (par exemple pour un bloc particulier)
) sera indiqué par un #

Par exemple :

```
#contenu {  
  position:absolute;  
  left:237px;
```

II) Structure d'un fichier CSS

```
top:55px;
bottom:30px;
right:0px;
overflow-y:auto;
font-size:1.2em;
}
```

une classe (par exemple pour une classe "générique") sera indiquée par un point "."

```
.repondre {
font-size:2em;
}
```

un élément possédant un identifiant et disposant en plus d'une classe sera indiqué comme ceci :

```
#contenu.cl_article {
right: 51px;
}
```

ici, le bloc identifié par contenu, s'il est en plus de classe cl_article, sera à 51px du bord droite de son conteneur.

Un élément disposant de plusieurs "versions" en fonction des événements sera indiqué par élément :événement.

Par exemple, pour un bloc à identifiant unique :

```
#zone_droite {
width:50px;
}

#zone_droite:hover {
width:300px;
}
```

Ce bloc fera 50px de large par défaut, mais lorsque la souris passera au-dessus du bloc, celui-ci s'élargira à 300px.

II) Structure d'un fichier CSS

C'est le cas du bloc situé à droite de cette page.

Cela permet de déplier/replier ou de rendre visible/invisible un élément facilement, juste avec quelques attributs de style, sans script compliqué.

A noter, certains navigateurs (la série des Microsoft Internet Explorer) ne gèrent pas le hover ailleurs que sur les liens.

On peut aussi définir un style commun à plusieurs éléments/identifiants/classes/événements. Ceux-ci seront séparés par des virgules.

Par exemple :

```
a.spip_out, a.spip_out:visited, a.spip_out:hover {
color: #42ADFB;
}

#entete, #pied {
width: 100%;
padding: 0px;
border: 0px;
}
```

Enfin, on peut aussi imbriquer des éléments/identifiants/classes/événements.

```
.menu h2 {
font-weight: bold;
text-align: center;
color:darkorange;
font-size:2.5em;
}
```

ici, on applique un style particulier aux éléments h2 situés dans un bloc de classe menu

```
#navigation a, #navigation a:hover, #navigation a:visited {
color: #999999;
}
```

II) Structure d'un fichier CSS

un style particulier aux liens situés dans le bloc d'identifiant navigation, quel que soit leur état (normal, survol, déjà visité)

```
#zone_droite:hover #zone_droite_contenu h2 {  
  font-size:2em;  
  line-height:2em;  
}
```

et là un style aux éléments h2 situés dans le bloc zone_droite_contenu lui-même situé dans le bloc zone_droite, lorsque celui-ci est survolé par la souris.