CSS

Feuilles de style en cascade

Introduction

Une page Web repose sur deux langages qui se partagent le travail. Le HTML dit **ce que contient** la page : un titre, des paragraphes, une liste, une image. Le **CSS** (*Cascading Style Sheets*, feuilles de style en cascade) dit **comment l'afficher** : couleurs, polices, tailles, marges, disposition à l'écran.

Cette séparation du fond et de la forme permet de changer toute l'apparence d'un site en modifiant un seul fichier CSS, sans toucher au contenu des pages, ou d'afficher la même page différemment sur un ordinateur et sur un téléphone. Apparu au milieu des années 1990, le CSS est normalisé, comme le HTML, par le W3C (*World Wide Web Consortium*), et tous les navigateurs actuels le comprennent.

Intégration de code CSS dans une page HTML

Les règles CSS peuvent être placées à trois endroits différents.

Première méthode (recommandée) : feuille de style externe

Les règles sont rangées dans un fichier à part, d'extension `.css` (par exemple `style.css`). Chaque page HTML indique dans son en-tête, avec une balise `<link>`, qu'elle utilise ce fichier :

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

Toutes les pages d'un site peuvent partager le même fichier : on écrit chaque règle une seule fois, et le navigateur, qui garde le fichier en mémoire après la première page, n'a plus besoin de le redemander au serveur.

Deuxième méthode : feuille de style interne

Les règles sont écrites dans la page elle-même, entre les balises `<style>` et `</style>`, dans l'en-tête :

```
<head>
    <style>
        body {
            background-color: linen;
        }
        h1 {
            color: maroon;
            margin-left: 40px;
        }
    </style>
```

```
</head>
```

Cette méthode convient à une page isolée, ou à quelques règles qui ne concernent qu'une seule page du site.

Troisième méthode (à éviter) : insertion directe dans les balises

La règle est écrite directement dans la balise concernée, grâce à l'attribut `style` :

```
<p style="color: red; font-weight: bold;">
    Ce paragraphe s'affiche en rouge et en gras.
</p>
```

On mélange alors à nouveau le fond et la forme : le HTML devient difficile à lire, et pour changer la couleur de tous les paragraphes il faudrait modifier chaque balise une par une.

Syntaxe

Une règle CSS commence par un **sélecteur**, qui désigne les éléments concernés. Viennent ensuite, entre accolades, des **déclarations** de la forme `propriété: valeur;`, une par ligne.

```
selecteur {
    propriete: valeur;
    propriete2: valeur2;
}
```

Remarque. Le point-virgule qui sépare deux déclarations est obligatoire : s'il manque, la déclaration se prolonge jusqu'au point-virgule suivant, sa valeur devient incompréhensible, et le navigateur ignore les deux déclarations. Les retraits et les retours à la ligne ne changent rien au résultat, mais rendent le code plus facile à relire.

Les sélecteurs

Pour préciser quelles balises doivent être mises en forme, on utilise des sélecteurs. Par exemple :

— `p` sélectionne toutes les balises paragraphe `<p>` ;

— `h1, h3` sélectionne toutes les balises `<h1>` et `<h3>` ;

— `body` sélectionne tout le contenu de la page web.

Le bloc de déclaration

Il s'agit des règles de style à appliquer sur les éléments sélectionnés. Par exemple :

```
p {
    color: red;
    text-align: center;
}
```

Cette règle colorie en rouge et centre tous les paragraphes du document HTML.

Les polices de caractères

La propriété `font-family` reçoit une liste de polices séparées par des virgules. Le navigateur utilise la première police de la liste qui est installée sur l'ordinateur du visiteur ; si aucune ne l'est, il se rabat sur le dernier mot-clé, qui désigne une famille générique :

— `serif` : une police avec **empattement**, c'est-à-dire avec les petits traits qui terminent les lettres (comme Times) ;

— `sans-serif` : une police sans empattement (comme Arial) ;

— `monospace` : une police à chasse fixe, dans laquelle toutes les lettres ont la même largeur (comme Courier).

```
body {
    font-family: "Helvetica Neue", Arial, sans-serif;
    font-size: 16px;
}
```

Commentaires

Les commentaires utilisent la syntaxe suivante : `/* Un commentaire */`

Sélecteurs plus élaborés

Il est possible de sélectionner plusieurs balises à la fois pour leur appliquer une règle commune ; ces balises doivent alors être séparées par des virgules :

```
h1, h2, h3 {
    color: lime;
}
```

La virgule est très importante, car si on sépare les sélecteurs par des espaces il s'agit d'une sélection par descendance. Le code CSS suivant ne sélectionne que les paragraphes contenus dans une balise `<div>` :

```
div p {
    background-color: red;
}
```

Pour appliquer un même style à plusieurs éléments, on peut leur ajouter un attribut HTML `class="ma-classe"`, sélectionné en CSS avec le sélecteur `.ma-classe`.

Pour appliquer un style spécifique à un unique élément, on lui ajoute un attribut HTML `id="mon-id"`, sélectionné en CSS avec le sélecteur `#mon-id`.

Classe ou id ?

La notion d'id (syntaxe : `#mon_id`)

Un `id` s'applique à un seul élément de la page. Exemple de fichier CSS :

```
#mon_par_de_police_rouge {
    color: red;
}

#mon_par_au_fond_bleu {
    background-color: cyan;
}
```

Et le fichier HTML correspondant :

```
<p id="mon_par_de_police_rouge">
Premier paragraphe en rouge.
</p>

<p>
Paragraphe normal.
</p>

<p id="mon_par_au_fond_bleu">
Paragraphe avec un fond bleu.
</p>
```

La notion de classe (syntaxe : `.ma_classe`)

Une `class` peut s'appliquer à plusieurs éléments. Exemple de fichier CSS :

```
.italique {
    font-style: italic;
}

.gras {
    font-weight: bold;
}

#mon_intro {
    text-transform: uppercase;
}
```

Et le fichier HTML correspondant :

```
<p class="italique" id="mon_intro">
Paragraphe en italique et en majuscules.
</p>

<p class="gras">
Paragraphe en gras.
</p>
```

```
8
9  <p class="italique gras">
10 Paragraphe en italique et en gras.
11 </p>
```

On remarque qu'un élément peut avoir plusieurs classes (séparées par des espaces) et qu'un `id` et une `class` peuvent coexister sur le même élément.

Ordre d'application des règles

Ce n'est pas pour rien qu'on parle de feuilles de styles *en cascade* : on peut définir des règles différentes pour des éléments identiques, et il faut savoir quelle règle s'applique.

Quand plusieurs règles s'appliquent au même élément, la plus **spécifique** l'emporte, par ordre croissant : règle sur un élément (`p`), puis sur une classe (`.note`), puis sur un identifiant (`#menu`), puis style en ligne dans la balise. À spécificité égale, c'est la dernière règle lue qui gagne, qu'elle vienne d'une feuille externe ou interne.

Déboguer son code CSS

Inspecteur de styles du navigateur

Pour savoir quelle règle a été utilisée par le navigateur, la console web est d'un grand secours. On peut inspecter un élément et voir quelles règles CSS s'appliquent. Les règles supplantées apparaissent barrées dans l'inspecteur de styles.

Validation du code CSS

Le site du W3C propose, comme pour le langage HTML, un validateur de fichier CSS.