

# Résumé HTML - CSS

qkzk

## Résumé des syntaxes HTML et CSS

### HTML

Les éléments d'une page HTML (fichier .html) sont *encapsulés* dans des balises

```
<balise> texte </balise>
```

Par exemple :

```
<h1>
  Ce texte est compris entre les balises h1, c'est un titre principal
</h1>
```

Ce texte n'est compris entre aucune balise. C'est une mauvaise pratique

```
<p>
  Un paragraphe avec
  <a href="http://qkzk.xyz/"> un lien absolu, </a>
  <a href="html_css/"> un lien relatif. Ce texte est dans plusieurs balises. </a>
</p>
```

Certaines balises ont des sémantiques particulières *qu'il faut connaître* :

```
<html> : le code html de la page elle même
<head> : l'entête
<body> : le corps

<link> : un lien vers un fichier, généralement de mise en page avec CSS
<h1>, <h2> les titres
<p> : un paragraphe
<a> : une "ancree" = un lien
<img> : une image
etc.
```

### CSS

La syntaxe du CSS est simple :

```
selecteur {
  propriete1: valeur1;
  propriete2: valeur2;
}
```

**selecteur** désigne certains éléments de la page (compris dans des balises html) Les propriétés "propriete1" et "propriete2" sont des propriétés du langage CSS. valeur1, valeur2 sont les attributs correspondants.

Il existe de nombreux moyens de sélectionner des éléments en CSS

```
/*
  Un Commentaire en CSS est compris entre ces deux "tags"
*/
```

Si le selecteur est noté sans aucune symbole le précédant il désigne TOUTES les balises en question. S'il est précédé d'un symbole `.` ou `#` il désigne respectivement une **classe** ou un **identifiant**. Les propriétés contenues dans les `{ }` seront appliquées à toutes les balises en question

Par exemple :

```
h1 {
  color: red;
}
/*
  Tous les titres h1 seront maintenant en rouge.
*/

.nomClasse {
  background-color: green;
}
/*
  On désigne une classe avec un .
  Tous les éléments ayant la CLASSE "nomClasse" auront un fond vert
*/

#identifiant {
  text-align: center;
}
/*
  On désigne une id avec un #
  L'unique élément de la page ayant l'id "identifiant" aura son texte centré
*/
```

## Selecteurs complexes

Il existe de nombreux moyens de sélectionner des parents, des enfants, de multiples classes ou combinaisons de classes.

Exemple tiré du CSS de mon site :

```
details[open] > summary:before {
  content: "-";
}
```

Cela permet d'avoir des `-` dans les menus sur lesquels on a cliqué plutôt que des `+`

- Le selecteur se comprend ainsi : “choisis tous les éléments `summary` qui sont contenus dans un élément détail qu'on a déjà ouvert `[open]` et, modifie ce qui est JUSTE AVANT (`:before`),”
- L'attribut `:content` désigne le contenu HTML, il devient `-`.

D'où le nom : selecteur *complexe*...

## Easy to learn, hard to master

- Tout le monde peut comprendre les rudiments de HTML et CSS,
- Tout le monde peut comprendre comment sont hiérarchisées les balises d'une page html,
- Maîtriser CSS et faire ce qu'on souhaite dans une page web est très difficile. C'est *une* compétence attendue du *métier* d'infographiste...

Je n'ai aucune prétention à ce sujet.

Un dernier exemple vous montrant ce que *j'ai* pu faire sur mon site en quelques jours :

- Je suis parti de ça
- Je voulais que ça ressemble à l'ancienne version de mon site
- Et le résultat, et bien vous êtes dessus.

Les deux versions se ressemblent assez pour qu'un élève connaissant la première s'y retrouve et pourtant elles n'ont rien à voir.

---

### Astuce dans sublime text ou atom (ou n'importe quel vrai éditeur de code).

Une fois votre fichier enregistré en *nom.html*, vous pouvez utiliser l'auto complétion :

Tapez `html` puis TAB et sublime text va générer tout le code suivant

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
</head>
<body>

</body>
</html>
```

Cela fonctionne avec toutes les balises courantes !

Essayez avec "a" puis TAB pour avoir :

```
<a href=""></a>
```

Cela *ne vous dispense pas* d'apprendre la syntaxe.