



CSS : Les Couleurs

Par Robert DIASSÉ

Chapitre CSS : Les Couleurs

La couleur est un élément crucial du design web, permettant d'ajouter de la vie et de l'attrait visuel à vos pages. En CSS, il existe plusieurs méthodes pour spécifier les couleurs et appliquer des effets spéciaux.

1. Utilisation des noms de couleur

Les noms de couleur prédéfinis sont des mots-clés simples qui représentent des couleurs spécifiques. Voici quelques exemples :

- `red` : rouge
- `blue` : bleu
- `green` : vert
- `yellow` : jaune
- `black` : noir
- `white` : blanc

2. Utilisation du code hexadécimal

Le code hexadécimal (0 à F) est une représentation numérique des couleurs en utilisant une combinaison de six chiffres hexadécimaux. Chaque paire de chiffres représente les valeurs de rouge, vert et bleu (RVB) communément `RGB` en anglais respectivement. Par exemple :

- `#FF0000` : rouge pur
- `#00FF00` : vert pur
- `#0000FF` : bleu pur

On peut aussi utiliser une version courte du code hexadécimal avec trois chiffres. Par exemple :

- `#F00` : rouge pur
- `#0F0` : vert pur
- `#00F` : bleu pur

Dans ce cas, chaque chiffre est répété pour représenter les valeurs RVB.

3. Utilisation de la fonction `rgb()`

La fonction `rgb()` permet de spécifier une couleur en utilisant les valeurs de rouge, vert et bleu dans une plage de 0 à 255. Par exemple :

- `rgb(255, 0, 0)` : rouge pur
- `rgb(0, 255, 0)` : vert pur
- `rgb(0, 0, 255)` : bleu pur

4. Utilisation de la fonction `rgba()`

La fonction `rgba()` est similaire à `rgb()`, mais elle permet également de spécifier une valeur d'opacité (alpha) pour la couleur. L'opacité est définie par un nombre compris entre 0 (complètement transparent) et 1 (complètement opaque). Par exemple :

- `rgba(255, 0, 0, 0.5)` : rouge pur avec une opacité de 50%

5. Utilisation de la fonction `hsl()`

La fonction `hsl()` permet de spécifier une couleur en utilisant les composantes teinte, saturation et luminosité. Les valeurs sont définies comme suit :

- Teinte (H) : une valeur entre 0 et 360 degrés représentant la couleur elle-même
- Saturation (S) : une valeur entre 0% (gris) et 100% (pleinement saturé)
- Luminosité (L) : une valeur entre 0% (noir) et 100% (blanc)

Par exemple :

- `hsl(0, 100%, 50%)` : rouge pur
- `hsl(120, 100%, 50%)` : vert pur
- `hsl(240, 100%, 50%)` : bleu pur

6. Dégradés et filtres de couleurs

En CSS, vous pouvez créer des dégradés de couleur avec les propriétés `linear-gradient()` et `radial-gradient()`. Ces dégradés permettent de passer en douceur d'une couleur à une autre selon un angle ou un rayon spécifié.

Les filtres CSS, tels que `brightness()`, `contrast()`, `blur()`, `sepia()` et bien d'autres, permettent d'appliquer des effets spéciaux aux couleurs. Par exemple, `brightness(200%)` augmente la luminosité de la couleur, tandis que `blur(5px)` ajoute un flou à la couleur.

```
.image-floue {  
  filter: blur(5px); /* Applique un flou de 5 pixels */  
}
```

Dégradé linéaire `linear-gradient()` :

La fonction `linear-gradient()` en CSS permet de créer un dégradé de couleur linéaire entre deux ou plusieurs couleurs, suivant une direction spécifiée. Voici comment elle fonctionne en détail :

Syntaxe de base :

```
linear-gradient(direction, color1, color2, ...);
```

- **direction** : Spécifie la direction du dégradé. Cela peut être défini en utilisant des valeurs d'angle (deg), des mots-clés (to top, to bottom, to left, to right, etc.) ou des pourcentages (par rapport à l'axe horizontal ou vertical).
- **color1, color2, ...** : Les couleurs entre lesquelles le dégradé sera créé. Vous pouvez spécifier autant de couleurs que vous le souhaitez.

Exemples :

1. Dégradé de haut en bas (du rouge au bleu) :

```
linear-gradient(to bottom, red, blue);
```

2. Dégradé de gauche à droite (du vert au jaune) :

```
linear-gradient(to right, green, yellow);
```

3. Dégradé en diagonale (du violet au cyan) :

```
linear-gradient(45deg, violet, cyan);
```

4. Dégradé horizontal répété (du rouge au jaune) :

```
linear-gradient(to right, red, yellow 50%, red);
```

Dans cet exemple, le dégradé va du rouge au jaune sur la première moitié, puis revient au rouge. Le dégradé va du rouge à partir du début de l'élément jusqu'à atteindre 50% de sa largeur, puis commence à se mélanger au jaune jusqu'à la fin de l'élément où il redevient rouge.

Fonctionnement :

- La fonction `linear-gradient()` crée une image de dégradé qui peut être utilisée comme valeur de propriété CSS. Cette image est ensuite appliquée comme arrière-plan à l'élément spécifié.
- Les couleurs spécifiées dans la fonction sont mélangées les unes aux autres le long de la ligne ou de l'axe défini par la direction, créant ainsi une transition en douceur d'une couleur à l'autre.

- Vous pouvez ajuster la répartition des couleurs en utilisant des pourcentages pour définir les points d'arrêt le long de l'axe. Cela permet de contrôler précisément où chaque couleur commence et se termine dans le dégradé.

La fonction `linear-gradient()` est utile pour créer des arrière-plans et des effets visuels intéressants sur les éléments HTML, comme des boutons, des bannières et des barres de navigation. Avec ses nombreuses options de personnalisation, elle offre une grande flexibilité pour réaliser des designs uniques et attrayants.

dégradé circulaire `radial-gradient()`

Le dégradé radial en CSS crée un effet de transition de couleur en partant d'un point central et en rayonnant vers l'extérieur dans toutes les directions. Voici comment il fonctionne en détail :

1. **Spécification de la forme** : La fonction `radial-gradient()` permet de spécifier un dégradé radial. Vous pouvez préciser la forme du dégradé en utilisant des valeurs telles que `circle` pour un cercle ou `ellipse` pour une ellipse.
2. **Position centrale** : Vous indiquez le point central du dégradé en utilisant les coordonnées x et y ou en pourcentage par rapport à l'élément parent. Cela se fait en utilisant la syntaxe `at <position>` où `<position>` peut être exprimée en pixels, en pourcentage ou en mots-clés (comme `center`, `top`, `left`, etc.).
3. **Couleurs du dégradé** : Vous spécifiez les couleurs qui composent le dégradé. Vous pouvez indiquer plusieurs couleurs et leurs positions dans le dégradé. Chaque position définit à quel point du rayonnement radial la couleur doit commencer à apparaître.
4. **Positions de couleur** : Les positions de couleur indiquent à quel point du rayonnement radial une couleur donnée commence à apparaître. Elles sont généralement exprimées en pourcentage de la distance radiale depuis le centre du dégradé.
5. **Combinaison** : En combinant la forme, la position centrale et les couleurs du dégradé, vous créez un effet visuel où les couleurs se mélangent à partir du point central et se propagent radialement vers l'extérieur.

Voici un exemple de syntaxe CSS pour un dégradé radial simple :

```
radial-gradient(circle at 50% 50%, red, yellow, green);
```

Dans cet exemple :

- `circle` indique une forme de cercle pour le dégradé.
- `at 50% 50%` définit le point central du dégradé au centre de l'élément.
- `red`, `yellow` et `green` sont les couleurs utilisées dans le dégradé, sans positions spécifiques, ce qui signifie qu'elles se fondent progressivement les unes dans les autres à partir du centre vers l'extérieur.

Propriétés CSS liées à la couleur

En plus des méthodes de spécification de couleur, voici quelques propriétés CSS importantes qui peuvent être utilisées pour manipuler les couleurs :

- `color` : spécifie la couleur du texte
- `background-color` : spécifie la couleur de fond d'un élément
- `border-color` : spécifie la couleur des bordures
- `box-shadow` : spécifie la couleur de l'ombre d'une boîte
- `text-shadow` : spécifie la couleur de l'ombre du texte

En combinant ces différentes méthodes et propriétés, vous pouvez créer une variété de styles et d'effets visuels pour vos éléments HTML. Amusez-vous à expérimenter avec les couleurs pour obtenir le look parfait pour votre site web.