

CSS : cascade, spécificité et héritage

Tristan Crolard

Laboratoire CEDRIC
Equipe « Systèmes Sûrs »

`tristan.crolard@cnam.fr`

`cedric.cnam.fr/sys/crolard`

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

Cascade et héritage

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade

[Aperçu : Building blocks](#) [Suivant](#)

Le but de cette leçon est de développer votre compréhension des concepts les plus fondamentaux du CSS — la cascade, la spécificité et l'héritage — qui contrôlent la manière dont le CSS est appliqué au HTML et la manière dont les conflits sont résolus.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé

Règles contradictoires

CSS est l'acronyme de **Cascading Style Sheets**, qu'on peut traduire par feuilles de style en cascade et la compréhension de ce premier mot cascading est cruciale — la façon dont la cascade se comporte est la clé de la compréhension du CSS.

À un moment donné, vous travaillerez sur un projet et vous constaterez que le CSS que vous pensiez appliquer à un élément ne fonctionne pas. En général, le problème est que vous avez créé deux règles qui pourraient potentiellement s'appliquer au même élément.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé

La cascade

À un niveau élémentaire, la **cascade** des styles signifie que l'ordre d'apparition des règles dans le CSS a une importance ; quand deux règles applicables ont la même spécificité, c'est la dernière déclarée qui sera utilisée pour la mise en forme.

Dans l'exemple ci-dessous deux règles pourraient s'appliquer à `h1`. Au final `h1` est coloré en bleu — ces règles ont les mêmes sélecteurs, elles ont donc la même spécificité ; dans ce cas, c'est la règle écrite en dernier dans le CSS qui l'emporte.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre
- Dans ce cours

This is my heading.

Interactive editor

```
h1 {
  color: red;
}
h1 {
  color: blue;
}
```

```
<h1>This is my heading.</h1>
```

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre

Spécificité

Quand des règles avec des sélecteurs différents s'appliquent sur un même élément, le navigateur choisit la règle qui a la plus grande spécificité. La spécificité mesure essentiellement combien la sélection est précise :

- Un sélecteur d'élément est peu spécifique — il cible tous les éléments d'un type donné dans la page — son score est donc faible ;
- Un sélecteur de classe est plus spécifique — dans la page, il ne cible que les éléments dont l'attribut `class` a la valeur choisie — son score est plus important.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre
- Dans ce cours

Related Topics

Bienvenue aux débutants !

Voyons cela sur un exemple. Ci-dessous, on retrouve deux règles qui pourraient s'appliquer à `h1`. Au final `h1` est coloré en rouge — le sélecteur de classe donne une plus grande spécificité à sa règle, et du coup c'est cette règle qui est choisie même si elle apparaît plus tôt dans le CSS.

This is my heading.

Interactive editor

```
.main-heading {
  color: red;
}
h1 {
  color: blue;
}
```

```
<h1 class="main-heading">This is my heading.</h1>
```

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé

Héritage

L'héritage est lui aussi à comprendre dans ce contexte — certaines valeurs pour une propriété CSS se transmettent des éléments parents vers leurs enfants, d'autres non.

Par exemple, si vous fixez une `color` et une `font-family` pour un élément, tout élément contenu dans le premier sera mis en forme avec la même couleur et la même police, à moins qu'on lui ait appliqué directement des règles.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre

Par exemple, si vous fixez une `color` et une `font-family` pour un élément, tout élément contenu dans le premier sera mis en forme avec la même couleur et la même police, à moins qu'on lui ait appliqué directement des règles.

As the body has been set to have a color of blue this is inherited through the descendants.

We can change the color by targeting the element with a selector, such as this span.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage

Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade

```
body {
  color: blue;
}

span {
  color: black;
}
```

<p>As the body has been set to have a color of blue this is inherited through the descendants.</p>
<p>We can change the color by targeting the element with a selector, such as this span.</p>

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires**
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre
- Dans ce cours

Certaines propriétés ne se transmettent pas — par exemple si vous attribuez un [width](#) de 50% à un élément, aucun de ses descendants n'aura une largeur diminuée de moitié par rapport à celle de son parent. Si c'était le cas, l'usage de CSS serait particulièrement frustrant !

Note : Sur MDN, dans les pages de référence des propriétés CSS, vous trouverez des encarts d'information technique, le plus souvent au pied de la section de spécifications, dans lesquels sont listés nombre de données sur la propriété, notamment si elle est héritée ou non. Voir la [section des spécifications de la propriété color](#), par exemple.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage

Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage**
- Comprendre la cascade

Dans la plupart des cas, le sens commun permet de comprendre quelles propriétés sont héritées par défaut et quelles propriétés ne se transmettent pas.

Contrôler l'héritage

CSS propose quatre valeurs spéciales universelles pour contrôler l'héritage. Ces valeurs sont applicables à toute propriété CSS.

<https://www.w3.org/TR/CSS22/propidx.html>

Principales propriétés héritées par défaut

- ▶ border-collapse
- ▶ border-spacing
- ▶ caption-side
- ▶ color
- ▶ font-family
- ▶ font-size
- ▶ font-style
- ▶ font-variant
- ▶ font-weight
- ▶ font (raccourci)
- ▶ letter-spacing
- ▶ line-height
- ▶ list-style-image
- ▶ list-style-position
- ▶ list-style-type
- ▶ list-style
- ▶ text-align
- ▶ text-indent
- ▶ text-transform
- ▶ visibility
- ▶ white-space
- ▶ word-spacing

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage**
- Comprendre la cascade
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre
- Dans ce cours

[inherit](#)

La propriété correspondante prend la valeur définie dans l'élément parent. Dans les faits, cela "active l'héritage".

[initial](#)

La propriété correspondante prend la valeur par défaut définie dans la feuille de style du navigateur. Si aucune valeur n'est définie par défaut dans le navigateur et que la propriété est transmise par héritage la propriété est redéfinie à `inherit`.

[unset](#)

Redéfinit la propriété à sa valeur naturelle : si la propriété est transmise par héritage, le comportement est le même que `inherit`, sinon il est identique à `initial`.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**
- Où sont écrites les règles CSS
- En résumé

Comprendre la cascade

Nous comprenons maintenant pourquoi un paragraphe imbriqué profondément dans la structure du code HTML a la même couleur que le `<body>`, et à partir des parties précédentes, nous comprenons comment changer la mise en forme d'un élément où qu'il soit dans le document — que ce soit par un sélecteur de type ou en créant une classe. Nous allons maintenant examiner comment la cascade détermine la règle CSS qui s'applique quand plusieurs règles ciblent le même élément.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**
- Où sont écrites les règles CSS
- En résumé

Il y a trois facteurs à prendre en compte, listés ci-dessous par ordre décroissant d'importance. Les premiers critères prennent le dessus sur ceux qui viennent après :

- 1. Importance**
- 2. Spécificité**
- 3. Ordre d'apparition dans le source**

Passons les en revue en partant de la fin, pour voir comment les navigateurs déterminent quel CSS doit être appliqué.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**
- Où sont écrites les règles CSS

Ordre d'apparition dans le source

Nous avons déjà vu comment l'ordre d'apparition dans le source compte dans la cascade. Si deux règles avec le même poids s'appliquent alors celle qui vient en dernier dans le CSS l'emporte. L'intuition est la suivante : plus on avance dans le CSS plus on s'approche de l'élément ciblé ; quand une règle le sélectionne, elle écrase la précédente jusqu'à la dernière règle rencontrée dans le source qui l'emporte et met en forme l'élément..

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**
- Où sont écrites les règles CSS
- En résumé
- Activité : jouer dans la cascade
- À suivre

Spécificité

L'ordre des règles dans le source est important. On rencontre pourtant des situations où deux règles ciblent le même élément mais c'est la première écrite dans le source qui s'applique. C'est que la première règle a une **spécificité** plus élevée — elle est plus spécifique, elle est donc choisie par le navigateur pour mettre en forme l'élément.

Comme nous l'avons vu plus haut dans cette leçon, un sélecteur de classe a plus de poids qu'un sélecteur d'élément, de sorte que les propriétés définies sur la classe remplaceront celles appliquées directement à l'élément.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**
- Où sont écrites les règles CSS

Un point important à noter : dans la cascade, on pourrait penser qu'une règle postérieure écrase une règle antérieure. En fait, ce n'est pas la règle toute entière qui est écrasée, mais seulement les propriétés communes aux deux règles qui sont redéfinies par la dernière version rencontrée.

Ce comportement permet d'éviter la répétition dans votre CSS. Une pratique courante consiste à définir des styles génériques pour les éléments de base, puis à créer des classes pour les cas particuliers.

The image shows a browser window with the address bar displaying 'developer.mozilla.org'. The page title is 'mdn web docs'. The breadcrumb navigation shows 'Apprendre le développement web > Cascade, spécificité et héritage'. The language is set to 'Français'. A sidebar on the left contains a 'Filter' button and a list of topics: 'Règles conflictuelles', 'Comprendre l'héritage', 'Comprendre la cascade' (highlighted), and 'Impact de l'emplacement'. The main content area contains the following text:

Voyons désormais comment le navigateur calcule la spécificité d'un sélecteur. Nous savons déjà qu'un sélecteur d'élément possède une spécificité inférieure à celle d'un sélecteur de classe. La spécificité est une valeur en points, associée à chaque type de sélecteur et la somme de ces poids fournit la spécificité d'un sélecteur composite donné. Cette mesure peut alors comparer aux autres.

developer.mozilla.org

mdn web docs

Blocs de base en CSS > Cascade, spécificité et héritage

Français

Filter

In this article

- Règles conflictuelles
- Comprendre l'héritage
- Comprendre la cascade**
- Impact de l'emplacement
- Évaluez vos compétences
- Résumé

Vous débutez ? Commencez ici !

- Démarrer avec le Web

La spécificité d'un sélecteur est mesurée selon 3 composantes différentes, qu'on peut voir comme des colonnes de centaines, dizaines et unités qui correspondent respectivement aux :

Identifiant

On marque un point dans cette colonne pour chaque sélecteur d'identifiant contenu dans le sélecteur composite.

Classe

On marque un point dans cette colonne pour chaque sélecteur de classe, d'attribut ou de pseudo-classe contenu dans le sélecteur composite.

Élément

On marque un point dans cette colonne pour chaque sélecteur d'élément ou de pseudo-élément contenu dans le sélecteur composite.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade, spécificité et héritage

Français

Filter

- Règles conflictuelles
- Comprendre l'héritage
- Comprendre la cascade**
- Impact de l'emplacement
- Évaluez vos compétences
- Résumé

Sélecteur	Identifiants	Classes	Éléments	Spécificité totale
<code>h1</code>	0	0	1	0-0-1
<code>h1 + p::first-letter</code>	0	0	3	0-0-3
<code>li > a[href*="en-US"] > .inline-warning</code>	0	2	2	0-2-2
<code>#identifiant</code>	1	0	0	1-0-0
<code>button:not(#mainBtn, .cta)</code>	1	0	1	1-0-1

Remarques :

- Dans Google Chrome, le triplet 1-0-2 est représenté (1,0,2)
- Par exemple, 1-0-2 est **plus spécifique** que 0-2-1 (car $102 \geq 21$)

A retenir

La déclaration appliquée à un élément est déterminée ainsi :

- ▶ Parmi les sélecteurs qui s'appliquent à l'élément, le **plus spécifique** est choisi.
- ▶ S'il en existe plusieurs avec la même spécificité, le **dernier** dans l'ordre d'inclusion des styles est choisi.
- ▶ S'il n'en existe aucun, mais que la **propriété est héritée**, on remonte dans l'arbre jusqu'à l'élément le plus bas qui a modifié cette propriété.
- ▶ Sinon, la valeur par défaut de la propriété est utilisée (style de l'agent).

The image shows a browser window displaying the MDN Web Docs page titled "Styles en incise dans le document". The browser's address bar shows "developer.mozilla.org". The page header includes the MDN logo, the text "Apprendre le développement web > Cascade, spécificité et héritage", and a language selector set to "Français". A left sidebar contains a "Filter" button and a list of navigation items: "Règles conflictuelles", "Comprendre l'héritage", "Comprendre la cascade" (highlighted), "Impact de l'emplacement", and "Évaluez vos compétences". The main content area features the title "Styles en incise dans le document" and a paragraph explaining that inline styles take precedence over all rules declared in style sheets, regardless of specificity, and are considered to have a specificity of 1-0-0-0.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade, spécificité et héritage Français

Filter

Règles conflictuelles

Comprendre l'héritage

Comprendre la cascade

Impact de l'emplacement

Évaluez vos compétences

Styles en incise dans le document

Les styles en incise du document (c'est-à-dire les déclarations de style présentes dans les attributs [style](#)) l'emportent sur toutes les règles déclarées dans les feuilles de style, quelle que soit leur spécificité. De telles déclarations n'utilisent pas de sélecteurs, mais on peut considérer leur spécificité comme 1-0-0-0, l'emportant ainsi toujours sur toute autre spécificité, quel que soit le nombre d'identifiants dans le sélecteur composite.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage

Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**

!important

Vous pouvez annuler tous les calculs ci-dessus à l'aide de la valeur CSS spéciale `!important` mais vous devez être très prudent avec son utilisation. Ce mot-clé est utilisé pour donner la plus grande spécificité à la propriété à laquelle il s'applique, remplaçant ainsi les règles normales de la cascade.

developer.mozilla.org

mdn web docs

Apprendre le développement web > Cascade et héritage

Français

In this article

- Règles contradictoires
- Comprendre comment ces concepts se combinent
- Comprendre l'héritage
- Comprendre la cascade**

Il est utile de connaître `!important`, ne serait-ce que pour le reconnaître dans le code des autres. **Cependant, nous vous recommandons fortement de ne jamais l'utiliser, sauf en dernier recours.** `!important` modifie le fonctionnement normal de la cascade, de sorte qu'il peut être très difficile de résoudre les problèmes de débogage CSS, en particulier dans une grande feuille de style.

https://developer.mozilla.org/fr/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance#!important