

CSS の優先順位について

作成日: 2016/01/21

作成者: 西村

更新履歴

更新日	更新概要	作業者
2016/01/21	・ 新規作成 (仮)	西村
2016/01/25	・ 再整理	西村
	・	
	・	
	・	

はじめに

CSS はどのスタイルを優先するか、という優先のルールが CSS の仕様書で決まっています。

覚えておくと、「なんで書いたスタイルが適用されないんだろう?」という疑問が少しわかり、スムーズに作業ができると思います。

この文書では、下記について説明します。

- ・ CSS のスタイルの優先順
- ・ 「詳細度」
- ・ 「継承」
- ・ 「!important」

問題

(1) この場合、の文字色は何色になるでしょう？ A.赤、B.青

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

(2) この場合、の文字色は何色になるでしょう？ A.赤、B.青

```
ul li {  
  color: red;  
}  
li {  
  color: blue;  
}
```

(3) この場合、の文字色は何色になるでしょう？ A.赤、B.青

```
ul li {  
  color: red !important;  
}  
ul li {  
  color: blue;  
}
```

(4) この場合、の文字色は何色になるでしょう？ A.赤、B.青

```
li {  
  color: red;  
}  
body {  
  color: blue;  
}
```

全問理由込みで分かる方はこの資料は読まなくてもよいかもしれません。

CSS の優先順位

CSS では、複数のルールからスタイルを適用されようとしている時(例えば同じタグに color を指定しているルールが 2 つあり、1 つは赤、1 つは青、になっている時)、下記の条件を持つものが優先されます。

1. 中に設定したスタイルが優先される
2. !important がついているスタイルが優先される
3. セレクタの詳細度が高いスタイル(指定の仕方が詳しいもの)が優先される
4. 後ろに書いてあるスタイルが優先される

1 が最も優先順位が高くなります。

※ 本来はもっとルール(ユーザー側のデフォルトスタイル等)がありますが、一般的に問題になることが少ないため省略しています。

優先度の低いものから順に説明していきます。

後ろに書いてあるスタイルが優先される

CSS は、基本的に後に書いてあるスタイルが勝つ、いわゆる後勝ちの方式を取っています。

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

同じもの(li)を 2 つ書いた場合あとの色が優先されるため、タグの文字は青になります。

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

これも同じで青になります。

セレクトタの詳細度が高いスタイルが優先される

詳細度(Specificity)は、「セレクトタがどのくらい詳しく持っているか」を示す度合いです。

※正式用語は「詳細度」ですが、「どのくらい特定しているか」という「特定度」と考えたほうがわかりやすいかもしれません。

セレクトタの例	意味
ul li	の中の
li	

「」より「の中の」のほうが詳しいので、「ul li」の方が詳細度が高い、ということになります。

そのため、下記ではタグの文字は赤になります。

```
ul li {  
  color: red;  
}  
li {  
  color: blue;  
}
```

詳細度には、もう少しルールがあります。「指定してあるものの種類によって詳しいポイントが違う」というものです。

#	項目
a	style 属性で設定されている(yes = 1, no = 0)
b	ID セレクトタ(#)が使われている数
c	クラスセレクトタ(.)や属性セレクトタ([type=text])が使われている数
d	要素セレクトタ(li)が使われている数

a > b > c > d の順で詳細度が高くなります。

(イメージ)

#	項目	イメージ
a	style 属性	タグに直接書き込んでいるため 「完全に特定している」
b	ID セレクトタ(#)	同じ ID 名は文書内で 1 つしか使えないため 「特定している度合いが高い」
c	クラスセレクトタ(.)や属性セレクトタ([type=text])	タグ(要素)に比べて 「特定している度合いが少し高い」
d	要素セレクトタ(li)	特定している度合いが低い

(例)

#	セレクタの例	style 属性	ID	クラス、属性等	要素(タグ)
1	*	0	0	0	0
2	ul li	0	0	0	2
3	li	0	0	0	1
4	html body ul li	0	0	0	4
5	.test	0	0	1	0
6	.test .item li	0	0	2	1
7	#test	0	1	0	0
8	(style 属性)	1	0	0	0

#4 のようにどれだけタグがたくさんあっても、#5 の方が詳細度が高いと判定されるので #4 が優先されます。

※ JavaScript でタグにスタイル設定をすると style 属性での設定と同等になるため、一番優先されることになります。

※ style 属性でのスタイル設定は仕事では使わないようにしてください。(JavaScript での設定除く)

※ style 属性についてはセレクタと直接関係ないことやそもそも非推奨なこともあり最近の仕様では詳細度の説明に style 属性を含めなくなっていますが、説明を簡単にするために style 属性を含めた説明にしています。

!important がついているスタイルが優先される

「!important」とは、「重要！」という意味の印です。

この印があると、詳細度が低かったり前に書かれていたりしている設定でも優先されます。

この印は、値の後ろに半角スペースで空けて書きます。(セミコロンよりは手前)

```
ul li {  
  color: red !important;  
}
```

下記のように本来なら青になるものでも、赤になります。

```
ul li {  
  color: red !important;  
}  
ul li {  
  color: blue;  
}
```

これは青です。(!important を付けなければ ul li の方が詳細度的に優先されますが、!important のため青になる)

```
ul li {  
  color: red;  
}  
li {  
  color: blue !important;  
}
```


中に設定したスタイルが優先される

これは<body>タグの文字が青に、タグの文字は赤になります。

```
li {  
  color: red;  
}  
body {  
  color: blue;  
}
```

<body>タグよりタグの方が中にあるためです。

```
<body>  
  こんにちは！ <!-- ここは青 -->  
  <ul>  
    <li>リスト項目</li> <!-- ここは赤 -->  
  </ul>  
</body>
```

CSS は、外のタグのスタイル設定が中に影響するプロパティと、しないプロパティがあります。

CSS の用語ではこの影響のことを「継承」(inheritance)と呼びます。

(継承するプロパティは文字関連が多く、文字色の color、文字サイズの font-size などが代表的です)

例えば上の例については li の color 設定がなければ body からの color の継承によりは青になります。

中のタグで新規に設定があれば、中のものが優先されます。(!important があろうとなかろうと、です)

補足

詳細度のルールは厳密に書くと下記ようになります。

#	項目
1	ID セクタ(#)が使われている数
2	クラスセクタ(.)や属性セクタ([type=text]), <u>擬似クラス(:first-child など)</u> が使われている数
3	要素セクタ(li), <u>擬似要素(::before など)</u> が使われている数

擬似クラスと擬似要素は別の資料で説明します。

参考文献

- Cascading Style Sheets Level 2 Revision 2 (CSS 2.2) Specification - 6 Assigning property values, Cascading, and Inheritance
 - <https://drafts.csswg.org/css2/>
- Cascading Style Sheets Level 2 Revision 2 (CSS 2.2) Specification 日本語訳 - 6 プロパティ値とカスケーディング、継承の割り当て
 - <http://momdo.github.io/css2/cascade.html>