

CSS のセレクトタについて

作成日: 2016/01/25

作成者: 西村

はじめに

CSS では、タグを指定するのに「セレクトタ」というものを使います。

セレクトタを覚えておくと、スタイルを付ける時に指定したいタグをうまく指定できるようになります。

JavaScript(jQuery)でもタグの指定の方法としてセレクトタを使うため、JavaScript も学びたい方は覚えておくとお得です。

この文書では、下記について説明します。

- ・ セレクトタ
- ・ 各セレクトタの意味
- ・ 擬似クラス
- ・ 擬似要素

セレクトア?

セレクトア(Selector。選択・指定するもの)は、どのタグにスタイルを付けるのかを指定するためのものです。

```
セレクトア {  
  プロパティ: 値;  
}
```

```
どのタグにスタイルを付けるかの指定 {  
  スタイルの設定名: スタイルの設定値;  
}
```

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



「body」を選択(指定)している

基本のセレクトク

いろいろありますが、まずは下記の4つが分かればいたいなとかなります。

(基本的なセレクトク)

やりたいこと	書き方	例
特定のタグ全部にスタイルを付けたい	タグ名を直接書く (「要素セレクトク」)	div → すべての<div>タグ
		p → すべての<body>タグ
		ul → すべてのタグ
特定のタグの中にあるタグにスタイルを付けたい	「タグ名 中のタグ名」のように、半角スペースで空けて書く (「子孫セレクトク」)	ul li → タグの中にあるタグ
		dl dd a → <dl>タグの中にある<dd>タグの中にある<a>タグ
特定の id を持つタグにスタイルを付けたい	「#ID 名」のように、半角のシャープのあとに ID 名を書く (「ID セレクトク」)	#test → id="test" が付いているタグ
特定の class を持つタグにスタイルを付けたい	「.クラス名」のように、半角のドットのあとにクラス名を書く (「クラスセレクトク」)	.test → class="test" が付いているタグ

※ それぞれの書き方を組み合わせることも出来ます。例えば「id="test"のタグの中の class="item"のタグ」であれば、「**#test .item**」とできます。

指定をシンプルにするコツ

スタイルを付けたいタグに ID やクラスをつけておくと、スタイルを付けるのがとても楽になります。

例えば、

```
<ul>
<li>1つ目のタグ</li>
<li>2つ目のタグ</li>
<li>3つ目のタグ</li>
<li>4つ目のタグ
  <ul>
    <li>4つ目の中の1つ目のタグ</li><!-- ← ここだけ色を赤くしたい! -->
    <li>4つ目の中の2つ目のタグ</li>
  </ul>
</li>
</ul>
```

ということになったとしたら、クラスか ID をつけてみてください。

```
<ul>
<li>1つ目のタグ</li>
<li>2つ目のタグ</li>
<li>3つ目のタグ</li>
<li>4つ目のタグ
  <ul>
    <li class="special">4つ目の中の1つ目のタグ</li><!-- ← ここだけ色を赤くしたい! -->
    <li>4つ目の中の2つ目のタグ</li>
  </ul>
</li>
</ul>
```

CSS 側は下記のようになります。

```
.special {
  color: red;
}
```

(class="special"がついているタグ、という意味のセレクトです)

もし付けなかった場合は、下記のようになります。

```
ul li ul li:first-child {
  color: red;
}
```

(ul の中の li の中の ul の中の li の最初のタグ、という意味のセレクトです。一気にややこしくなります)

※id は同じページ内の複数のタグに同じ名前で行けることは許されていません。複数のタグに同じ名前を付けたい場合はクラスを使いましょう。

「もっといろいろ覚えたい」という方は次のページ以降も見てみてください。
(前に挙げた 4 つのセレクトが使いこなせるようになってからで構いません)

関係性を示すセレクト

「ul タグの中の li」など、構造上の関係を指定したい場合は下記のようなセレクトを使います。(よく使うのは 1 つ目)

パターン	意味	名称
E F	E の中に含まれる F タグ	子孫セレクト (Descendant combinator)
E > F	E の直下に含まれる F タグ (E タグの子供である F タグ)	子供セレクト (Child combinator)
E + F	兄弟として E のすぐ後ろにある F タグ (E と F は同じタグの兄弟)	隣接セレクト (Adjacent sibling combinator)
E ~ F	兄弟として E の後ろにある F タグ (E と F は同じタグの兄弟)	兄弟セレクト (General sibling combinator)

子供？子孫？

```
<body>
  <ul>
    <li>項目 1</li>
    <li>項目 2</li>
  </ul>
</body>
```

の場合

- ・ は<body>の子供であり子孫です (**body ul** や **body > ul** で指定可)
- ・ は<body>の子孫です (**body li** で指定可。子供(直下)にないので **body > li** は不可)

隣接？兄弟？

```
<ul>
  <li>項目 1</li>
  <li id="test2">項目 2</li>
  <li>項目 3</li>
  <li>項目 4</li>
</ul>
```

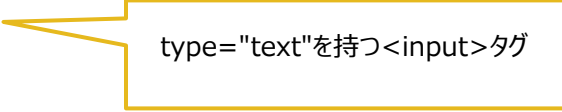
の場合

- ・ はすべての子供(直下)なので兄弟です。
- ・ 「**#test2 + li**」は「兄弟として #test2 のすぐ後ろにある li タグ」を示すので、「項目 3」のみ指定されます。
- ・ 「**#test2 ~ li**」は「兄弟として E の後ろにある li タグ」を示すので、「項目 3」「項目 4」が指定されます。

属性セクタ

タグだけでなく、「属性として何を持っているか」というところまで指定したい時には「属性セクタ」を使うことができます。

```
input[type="text"] {  
  color: red;  
}
```



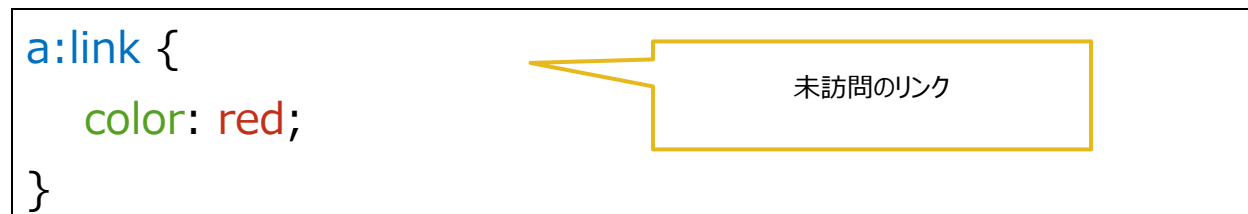
(例)

パターン	意味	名称
E[foo]	"foo"という属性を持っている E タグ	属性セクタ (Attribute selectors)
E[foo="bar"]	foo="bar"という属性を持っている E タグ	
E[foo~="bar"]	foo 属性の値(空白区切り)の一部として、"bar"という文字がある E タグ (foo="foo bar baz"など)	
E[foo^="bar"]	foo 属性の値が"bar"という文字から始まる E タグ	
E[foo\$="bar"]	foo 属性の値が"bar"という文字で終わる E タグ	
E[foo*="bar"]	foo 属性の値に"bar"という文字が含まれる E タグ	
E[foo = "en"]	foo 属性の値(ハイフン区切り)の一部として、"en"という文字がある E タグ (foo="en-US"など)	

擬似クラス

「擬似クラス」(pseudo-class。しゅーどくらす※)は、「状態を擬似的なクラス(分類)として指定できるもの」です。

よくあるのは「未訪問のリンク」を指定する「a:link」、「訪問済のリンク」を示す「a:visited」です。



セレクタ	意味	名称
E:link	E タグがリンクで、利用者が未訪問のリンク	リンク擬似クラス (The link pseudo-classes)
E:visited	E タグがリンクで、利用者が訪問済のリンク	

「マウスオーバー時(:hover)」「フォーカス時(:focus)」といった状態も指定できます。

パターン	意味	名称
E:hover	マウスオーバー時の E タグ	アクション擬似クラス (The user action pseudo-classes)
E:focus	フォーカス時の E タグ	
E:active	ユーザーに選択されている時の E タグ	

また、「最初の子供(:first-child)」「最後の子供(:last-child)」といった「構造がどうなっているか」、という状態での指定もできます。

パターン	意味	名称
E:first-child	親のタグから見て、最初の子供である E タグ	構造的擬似クラス (Structural pseudo-classes)
E:last-child	親のタグから見て、最後の子供である E タグ	

「セレクタの一覧」の「構造的擬似クラス」にその他のものも掲載しています。

※ pseudo の p は psycho-(サイコ-)と同じく発音しない

擬似要素

「擬似要素」は、「タグとしては存在しないものに対してタグ(要素)のようにスタイルを付けることができるもの」です。

例えば「タグの前や後にテキストを追加する(::before, ::after)」「タグの 1 文字目だけ赤くする(::first-letter)」などができます。

```
a::before {
  content: "test";
  color: red;
}
```

a の前に赤色で「test」を挿入

```
p::first-letter {
  font-size: 30px;
  font-weight: bold;
}
```

p(段落)の 1 文字目だけ大きく

パターン	意味	名称
E::before	E タグの前	::before 擬似要素 (The ::before pseudo-element)
E::after	E タグの後ろ	::after 擬似要素 (The ::after pseudo-element)
E::first-line	E タグの 1 行目	::first-line 擬似要素 (The ::first-line pseudo-element)
E::first-letter	E タグの 1 文字目	::first-letter 擬似要素 (The ::first-letter pseudo-element)

※擬似要素はコロン 1 つでも動きます。昔の CSS では擬似クラスも擬似要素もコロン 1 つだったためです。

セクタの一覧

セクタの一覧は、「Selectors Level 3」という仕様で決まっています。※
仕様書は英語なので、意訳したものを掲載します。(全部覚える必要はありません)

- ・ ここでの「E」や「foo」は HTML には存在しない、例のためのタグ名(要素名)や属性名です。「E」は「div」「p」など、「foo」は「href」などに置き換えて読み進めて下さい。
- ・ よく使うものを上に持って行くなど、仕様書とは順番を変更しています。

※ 「CSS 1」→「CSS 2.1」→「Selectors Level 3」→「Selectors Level 4」という形でアップデートされていっていますが、レベル 4 は 2016/01 現在ではまだ正式なものではないのでレベル 3 のものを掲載します。

基本的なセクタ

基本になるセクタです。

パターン	意味	名称
*	すべての要素	全称セクタ (Universal selector)
E	E というタグ(要素)	要素セクタ または 型セクタ (Type selector)

クラス・ID セクタ

「どのクラス、ID を持つか」を指定したい時に使います。

パターン	意味	名称
E.warning	"warning"というクラスを持つ E タグ (class="warning" や class="warning bar")	クラスセクタ (Class selectors)
E#myid	"myid"という id を持つ E タグ	ID セクタ (ID selectors)

※ クラスや ID セクタは、「.warning」や「#myid」のようにタグ名を指定しないことが多いです。

コンビネータ(関係性を示すもの)

「どのタグの中にあるタグか」などを指定したい時に使います。

パターン	意味	名称
E F	E の中に含まれる F タグ	子孫セクタ (Descendant combinator)
E > F	E の直下に含まれる F タグ	子供セクタ (Child combinator)
E + F	E のすぐ後ろにある F タグ (E と F は同じタグの兄弟)	隣接セクタ (Adjacent sibling combinator)
E ~ F	E の後ろにある F タグ (E と F は同じタグの兄弟)	兄弟セクタ (General sibling combinator)

属性セクタ

「属性として何を持っているか」というところまで指定したい時に使います。

パターン	意味	名称
E[foo]	"foo"という属性を持っている E タグ	属性セクタ (Attribute selectors)
E[foo="bar"]	foo="bar"という属性を持っている E タグ	
E[foo~="bar"]	foo 属性の値(空白区切り)の一部として、"bar"という文字がある E タグ (foo="foo bar baz"など)	
E[foo^="bar"]	foo 属性の値が"bar"という文字から始まる E タグ	
E[foo\$="bar"]	foo 属性の値が"bar"という文字で終わる E タグ	
E[foo*="bar"]	foo 属性の値に"bar"という文字が含まれる E タグ	
E[foo ="en"]	foo 属性の値(ハイフン区切り)の一部として、"en"という文字がある E タグ (foo="en-US"など)	

リンク擬似クラス

未訪問のリンク(「a:link」)、訪問済のリンク(「a:visited」)で色を変えたい場合などに使います。

セレクタ	意味	名称
E:link	E タグがリンクで、利用者が未訪問のリンク	リンク擬似クラス (The link pseudo-classes)
E:visited	E タグがリンクで、利用者が訪問済のリンク	

ユーザーアクション擬似クラス

「選択されている」「マウスオーバー時」「フォーカス時」などの状態によって色を変えたい場合などに使います。

パターン	意味	名称
E:hover	マウスオーバー時の E タグ	アクション擬似クラス (The user action pseudo-classes)
E:focus	フォーカス時の E タグ	
E:active	ユーザーに選択されている時の E タグ	

UI 状態擬似クラス

「有効」「無効」「チェックされている」などの状態によってスタイルを変えたい場合などに使います。

パターン	意味	名称
E:enabled	有効(enabled)な E タグ	:enabled 擬似クラス
E:disabled	無効(disabled)な E タグ (disabled 属性が付いているなど)	または UI 状態擬似クラス (The UI element states pseudo-classes)
E:checked	チェックされている E タグ (ラジオボタンやチェックボックス)	:checked 擬似クラス または UI 状態擬似クラス (The UI element states pseudo-classes)

その他の擬似クラス

パターン	意味	名称
E:target	ターゲットになっている E タグ	ターゲット擬似クラス (The target pseudo-class)
E:lang(fr)	lang 属性が"fr"の E タグ	:lang()擬似クラス (The :lang() pseudo-class)
E:not(s)	s の条件に一致しない E タグ	:not()擬似クラス または否定擬似クラス (Negation pseudo-class)

構造的擬似クラス

「タグの中の最初の子供」(:first-child)など、HTML の構造によってスタイルを変えたい場合に使用します。

パターン	意味	名称
E:first-child	親のタグから見て、最初の子供である E タグ	構造的擬似クラス (Structural pseudo-classes)
E:last-child	親のタグから見て、最後の子供である E タグ	
E:first-of-type	親のタグから見て、E タグだけで見て最初の子供である E タグ	
E:last-of-type	親のタグから見て、E タグだけで見て最後の子供である E タグ	
E:nth-child(n)	親のタグから見て、順番としてタグの種類問わず n 番目の子供である E タグ	
E:nth-last-child(n)	親のタグから見て、順番としてタグの種類問わず後ろから n 番目の子供である E タグ	
E:nth-of-type(n)	親のタグから見て、E タグだけで数えて n 番目の子供である E タグ	
E:nth-last-of-type(n)	親のタグから見て、E タグだけで数えて後ろから n 番目の子供である E タグ	
E:only-child	親のタグから見て、唯一のタグである E タグ	
E:only-of-type	親のタグから見て、E タグだけで見て唯一のタグである E タグ	
E:empty	中に子供を持たない E タグ	
E:root	ルートになっている E タグ	

擬似要素

タグとしては存在しないものに対してタグのようにスタイルを付けることができるものです。

パターン	意味	名称
E::before	E タグの前	::before 擬似要素 (The ::before pseudo-element)
E::after	E タグの後ろ	::after 擬似要素 (The ::after pseudo-element)
E::first-line	E タグの 1 行目	::first-line 擬似要素 (The ::first-line pseudo-element)
E::first-letter	E タグの 1 文字目	::first-letter 擬似要素 (The ::first-letter pseudo-element)