

擬似クラスとアニメーション

擬似クラスとは
要素がある状態になった時にスタイルを適用させる。

- `:link` リンクのデフォルトの状態
- `:visited` 既にクリックされたことがある(訪問されている)状態
- `:hover` マウスをリンクの上に置いた時の状態
- `:active` 擬似クラスはリンクがアクティベート(活性化)された時の状態

`:visited` 擬似クラスによって定義されたスタイルは、擬似クラス (`:link`, `:hover`, `:active`) によって上書きされます。適切にリンクにスタイルを適用するには、
LVHA 順、`:link` - `:visited` - `:hover` - `:active` に記述します。

擬似クラスの使い方

セレクタ:擬似クラス{

その状態になった時に適用させたいスタイルを記入

}

hoverのアニメーション例 1

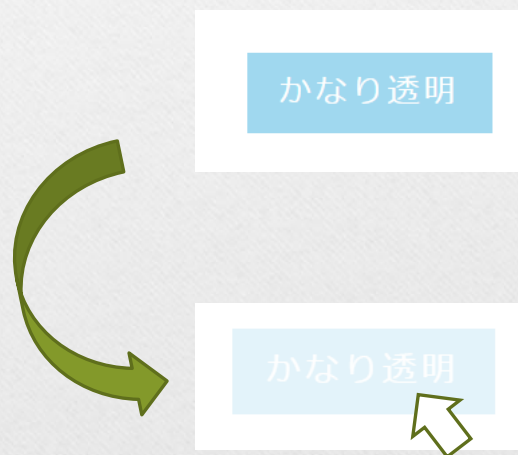
1.【透けさせる】opacity

HTML

```
<a href="xxx.html">かなり透明</a>
```

CSS

```
a {  
  background: skyblue;  
  padding: 8px 16px;  
  color: white;  
}  
a:hover {  
  opacity: 0.3;  
}
```



hoverのアニメーション例 2

2.【角を丸める】border-radius

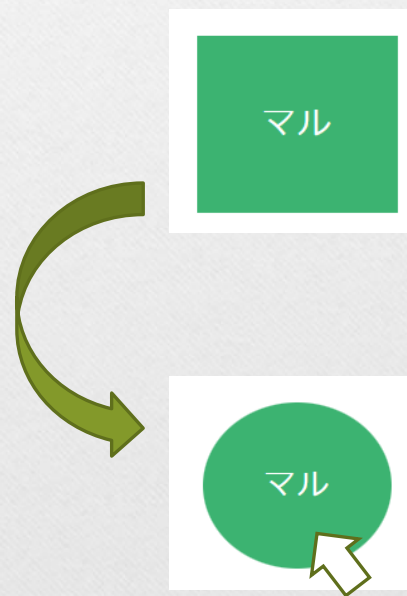
HTML

```
<a href="xxx.html">マル</a>
```

CSS

```
a {  
  background: mediumseagreen;  
  padding: 32px;  
  color: white;  
}
```

```
a:hover {  
  border-radius: 50%;  
}
```



hoverのアニメーション例 3

3.【移動・回転・拡大&縮小など】transform

■ 移動

```
セレクト {  
transform: translate(右方向○px, 下方向○px);  
}
```

[例]

```
div {  
transform: translate(5px, 0);  
}
```

■ 回転

```
セレクト {  
transform: rotate(○deg);  
}
```

[例]

```
div {  
transform: rotate(45deg);  
}
```

■ 拡大&縮小

```
セレクト {  
transform: scale(縦○倍, 横○倍);  
}
```

[例]

```
div {  
transform: scale(0.5, 1.5);  
}
```


3.1 【拡大&縮小】transform: scale(n, n)記述例

拡大&縮小

HTML

```
<div class="trans-scale">ちいさくなる</div>
```

CSS

```
.trans-scale {  
  background: salmon;  
  padding: 16px;  
  color: white;  
}  
  
.trans-scale:hover {  
  transform: scale(.7, .7);  
}
```



ちいさくなる

ちいさくなる



hoverのアニメーション例 4

4.1 Transition

ボタンにマウスカーソルを乗せたときに、色や大きさなどが**変わる時間を設定**するときに使われるプロパティ

プロパティ名	読み方	説明
transition-duration	トランジション・デュレーション	変化が始まって終わるまでの時間を指定します。
transition-property	トランジション・プロパティ	変化が適用されるCSSのプロパティを指定します。
<u>transition-timing-function</u>	トランジション・タイミング・ファンクション	変化の度合いを指定します。
transition-delay	トランジション・ディレイ	変化が始まる時間を指定します。(初期値0s)
transition	トランジション	上記の4つのプロパティを一括で指定できる、ショートハンド・プロパティです。

4.2 transition-timing-function

アニメーションが開始され終了するまでにおける変化の進行度合いを定義します。

値	説明
ease	開始時と終了時にはゆっくり変化（初期値）
ease-in	開始時はゆっくり変化し、終了時は早く変化
ease-out	開始時に早く、終了時にゆっくり変化
ease-in-out	easeよりさらに、開始時と終了時はゆっくり変化
linear	速度が変わることなく一定に変化
cubic-bezier(x軸の値, y軸の値, x軸の値, y軸の値)	変化の度合いをベジェ曲線により任意で指定

4.3 transition-timing-function

記述例

HTML

```
<div class="button">ホバーで色が変わります</div>
```

CSS

```
.button {  
    padding: 10px;  
    color: #ffffff;  
    background-color: #1A60BF;  
    transition-property: all;  
    transition-duration: 0.5s;  
    transition-delay: 0s;  
    transition-timing-function: ease-in;  
}  
  
.button:hover{  
    color: #1A60BF;  
    background-color: #D0E2FB;  
}
```

/* 一括指定 */

transition: all 0.5s 0s ease-in;

4.5 Transition(一括指定)

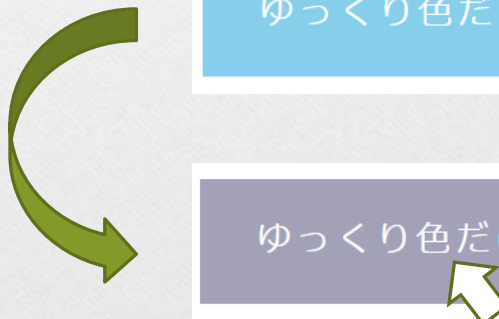
- "セレクトタ"から"セレクトタ: hover"までの変化の秒数を決める
- **hover**に書いたプロパティが変化する
- 変化の対象にallと書くと、:hoverの全てのプロパティを変化対象にできる。

HTML

```
<div class="only-background">ゆっくり色だけ変化</div>
```

CSS

```
.only-background {  
  background: skyblue;  
  padding: 16px;  
  color: white;  
  transition: background 1s;  
}  
.only-background: hover {  
  background: #669;  
}
```



Allを指定するとすべてのプロパティを変化させることができます。

Activeのアニメーション例

【影を出す】box-shadow

セレクトタ {

box-shadow: 0px (右方向の影) 0px (下方向の影) 0px (影のぼかし具合) 色;
}

[例] div { box-shadow: 5px 0 3px red; }

HTML

<div class="bottom-down">ボトムダウン</div>

CSS

```
.bottom-down {  
  background: mediumseagreen;  
  color: white;  
  box-shadow: 0 5px black;  
}  
.bottom-down:active {  
  box-shadow: none;  
  transform: translate(0, 5px);  
}
```

ボトムダウン

ボトムダウン

pxをマイナスで指定すると、左や上方向にも影を作れます。

完成例 (x100_MyPortfolio_作成日¥works.html) x100_MyPortfolioをコピーして作成

アニメーションデザイン例

1. グリッドアイテムに課題名は表示されません。



2. アイテムにマウスを乗せるとゆっくりと回転しながら背景と文字、枠線が浮き上がります。



3. 既にリンク済みの場合は文字の色が変わります。



サンプルコード

```
/*=====
|      | Animation
=====*/
#works a{
    opacity: 0; /* 非表示する */
    transition: all 3s; /* 全てのプロパティを3秒かけて変化 */
    transform: rotate(0.5turn) scale(0); /* 文字を反転 拡大率0 */
}

#works a:visited{ /* リンク済み */
    color: □white;
}

#works div:hover a{ /* hoverのイベントをdivでとる */
    opacity: 1; /* 表示する */
    border:solid ■rgb(129, 41, 88) 3px;
    transform: rotate(1turn) scale(1); /* 文字を正転 拡大率1 */
    background-color: □beige;
}
```

リンクの状態によってデザインを変えましょう。
デザインは自由です。
セレクトタに疑似クラス、CSSプロパティに
transition
transform
opacityなどを利用してください。

サンプルコードと同じ
デザインでは評価
値が下がります。