

Animation -transition 編 -

-HTML -

```
<body>  
  <div>button</div>  
</body>
```

-CSS-

```
div {  
  width: 150px;  
  line-height: 50px;  
  background: khaki;  
  text-align: center;  
}  
div:hover {  
  background: purple;  
  color: white;  
}
```

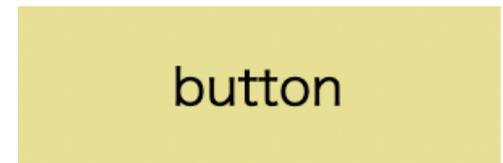
通常



- CSS transition 設定 -

```
div {  
  transition-property: background;  
  transition-duration: 3s;  
  transition-timing-function: ease-in-out;  
  transition-delay: 1s;  
}
```

transition 設定



- まとめて記入することも可能 -

```
div {  
  transition: 3s background ease-in-out 1s;  
}
```

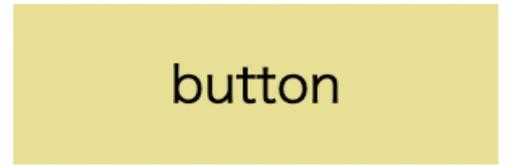


hover すると 1s 経過後に 3s かけて
css の内容が変更される

-CSS に追加 -

```
div:hover {  
  width: 200px;  
  line-height: 70px;  
  border-radius: 50%;  
  transform: rotate(10deg) translate(50px,30px);  
  font-weight: bold;  
}
```

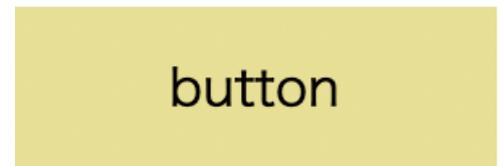
通常



-CSS transition 設定 -

```
div {  
  transition-property: all;  
  transition-duration: 3s;  
  transition-timing-function: ease-in-out;  
  transition-delay: 1s;  
}
```

transition 設定



- まとめて記入することもできる -

```
div {  
  transition: 3s all ease-in-out 1s;  
}
```

hover すると 1s 経過後に
3s かけて内容が変更される

- **transition-property** トランジション効果を適用するプロパティを指定 -
指定できる値

- ・ プロパティ名 適用する css のプロパティ名を指定。
複数のプロパティを指定する時は (,) で区切る。
- ・ all 適用できる全てのプロパティを指定。
- ・ none 全てのプロパティを適用しない。

- **transition-duration** トランジション効果が完了するまでの時間を指定 -
指定できる値

- ・ 数値+単位 (初期値は0) s (秒) ms (ミリ秒)
複数の時間を指定する時は (,) で区切る。

- **transition-timing-function** トランジション効果の変化のタイミングを指定 -
指定できる値

- ・ ease(初期値) 開始と完了を滑らかな変化にする。
- ・ linear 直線的に変化する。
- ・ ease-in ゆっくり開始する。
- ・ ease-out ゆっくり終了する。
- ・ ease-in-out ゆっくり開始し、ゆっくり終了する。
- ・ cubic-bezier 関数型で変化のタイミングを指定する。
- ・ step-start 開始から終了状態に一気に変化し、その後は変化しない。
- ・ step-end はじめは変化せず、最後に終了状態に一気に変化する。
- ・ steps 等時間で区切ったステップ数で変化する。
1つ目の値にステップ数、2つ目の値に start、end のいずれかを指定する。

- **transition-delay** トランジション効果が開始されるまでの時間を指定 -
指定できる値

- ・ 数値+単位 (初期値は0) s (秒) ms (ミリ秒)
複数の時間を指定する時は (,) で区切る。

- **transition** トランジション効果をまとめて指定 -
記述順序は任意それぞれスペースで区切って記入。

ただし、transition-duration transition-delay 2つを指定する場合は、transition-duration を先に記入する、

Animation -keyframes01 編 -

-HTML -

```
<body>
  <div>button</div>
</body>
```

-CSS-

```
div {
  width: 200px;
  line-height: 30px;
  background: pink;
  text-align: center;
  color: red;
}
```



- @keyframes 規則 アニメーションの動きとタイミングを指定 -

指定できる値

- ・アニメーション名 animation-name プロパティでしたアニメーション名を記述。
これにより、@keyframes 規則とアニメーションを指定する要素が結びつく。
- ・% %で変化させるタイミングを指定する。
- ・from アニメーションの開始時点を指定する。(0%と同じ)
- ・to アニメーションの終了時点を指定する。(100%と同じ)

- CSS keyframes設定 -

```
@keyframes anime01 {
  0% {
    width: 100px;
    line-height: 30px;
    background: olive;
    color: white;
  }
  100% {
    width: 300px;
    line-height: 70px;;
    background: lavender;
    color: tomato;
    font-size: 2em;
  }
}
```



-CSS animation 設定 -

```
div {  
    animation-name: anime01;  
    animation-duration: 5s;  
    animation-delay: 2s;  
    animation-timing-function: ease-in-out;  
    animation-fill-mode: both;  
    animation-iteration-count: 3;  
    animation-direction: alternate;  
}
```

-CSS animation 設定 -

```
div:hover {  
    animation-play-state: paused;  
}
```



hover すると、
アニメーションが一時停止される

- animation-name 要素にアニメーション名を指定 -

指定できる値

- ・ none (初期値) アニメーション名を指定しない。(アニメーションは実行されない)
- ・ アニメーション名 @keyframes 規則のアニメーション名を指定する。
複数のアニメーション名を指定する時は、(,) 区切る。

- animation-duration アニメーションの1回分の時間を指定 -

指定できる値

- ・ 数値+単位 (初期値は0) s (秒) ms (ミリ秒)
複数の時間を指定する時は (,) で区切る。

- animation-delay アニメーションが開始するまでの時間を指定 -

指定できる値

- ・ 数値+単位 (初期値は0) s (秒) ms (ミリ秒)
複数の時間を指定する時は (,) で区切る。

- animation-play-state アニメーションが再生中か一時停止状態かを指定 -

指定できる値

- ・ running (初期値) 一時停止中のアニメーションを再生する。
- ・ paused 再生中のアニメーションを一時停止する。

- animation-timing-function アニメーションの変化のタイミングを指定 -

指定できる値

- ・ ease(初期値) 開始と完了を滑らかな変化にする。
- ・ linear 直線的に変化する。
- ・ ease-in ゆっくり開始する。
- ・ ease-out ゆっくり終了する。
- ・ ease-in-out ゆっくり開始し、ゆっくり終了する。
- ・ cubic-bezier 関数型で変化のタイミングを指定する。
- ・ step-start 開始から終了状態に一気に変化し、その後は変化しない。
- ・ step-end はじめは変化せず、最後に終了状態に一気に変化する。
- ・ steps 等時間で区切ったステップ数で変化する。

1つ目の値にステップ数、2つ目の値に start、end のいずれかを指定する。

- animation-fill-mode アニメーションの実行前後のスタイルを指定 -

指定できる値

- ・ none(初期値) アニメーション実行の前後にスタイルを指定しない。
実行後は元のスタイルを適用。
- ・ forwards アニメーション終了時に最後のキーフレーム(100%)のスタイルを適用。
- ・ backwards アニメーション開始時に animation-delay プロパティが指定されている場合も、最初のキーフレーム(0%)のスタイルを適用。
- ・ both アニメーション開始時は backwards、終了時は forwards と同じスタイルを適用。

none



forwards



backwards



both



- **animation-iteration-count** アニメーションの実行回数を指定 -

指定できる値

- ・ 数値（初期値 1） アニメーションを何回実行するかを数値で指定。
1.5 のように小数で指定すると途中まで再生される。
- ・ infinite アニメーションを制限なく実行し続ける。

- **animation-direction** アニメーションの再生方向を指定 -

指定できる値

- ・ normal（初期値） 通常の方法に再生される。
- ・ reverse 逆方向に再生される。
- ・ alternate アニメーションの繰り返し回数が奇数の時は通常の方法に、偶数の時は逆方向に再生される。
- ・ alternate-reverse アニメーションの繰り返し回数が奇数の時は逆方向に、偶数の時は通常の方法に再生される。

- **animation** アニメーション関連のプロパティをまとめて指定 -

記述順序は任意それぞれスペースで区切って記入。

ただし、animation-duration animation-delay 2 つを指定する場合は、animation-duration を先に記入する、

Animation -keyframes02 編 -

-HTML -

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>Keyframes02</title>
</head>
<body>
  <p></p>
</body>
</html>
```

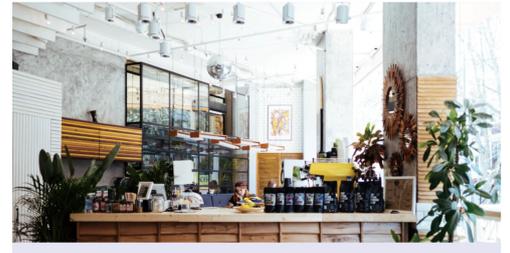
-CSS-

```
img {
  width: 100%;
  height: 100%;
  object-fit: cover;
}
p {
  width: 200px;
  background: pink;
  text-align: center;
}
```



-CSS-

```
@keyframes anime01 {  
  0% {  
    background: olive;  
    transform: scale(0.5,0.5) rotate(-20deg);  
  }  
  100% {  
    background: lavender;  
    transform: scale(3,3) translate(40%,50px);  
  }  
}
```



-CSS-

```
p {  
  animation-name: anime01;  
  animation-duration: 5s;  
  animation-delay: 1s;  
  animation-timing-function: ease-in-out;  
  animation-fill-mode: backwards;  
  animation-iteration-count: 3;  
  animation-direction: alternate-reverse;  
}
```

