

CSS ANIMATION

CSS Animation

```
@keyframes nazwaAnimacji {  
    0% { width: 100%; color: #ee4388; }  
    50.00% { width: 10%;}  
    50.01% {width: 10%}  
    100% { width: 100%; #ee4388;}  
}
```

```
div { animation: nazwa 10s }
```

```
animation-name: nazwa; /* nazwa której użyliśmy w @keyframes */  
animation-duration: 3.2s;
```

Dwie wymagane wartości.

Wartości domyślne:

```
animation-name: none;  
animation-duration: 0;
```

animation-delay

animation-delay: 2s;

Opóźnienie względem uruchomienia animacji. Jeśli animacja ma być uruchomiona 2 sekund po wczytaniu strony to ustawiamy "2s". Po najechaniu na element (:hover), po kliku czy skrołu (JavaScript).

animation: superAnimacja 10s 2s

Jeśli nie użyjemy tej właściwości to domyślna wartość to 0;

animation-direction

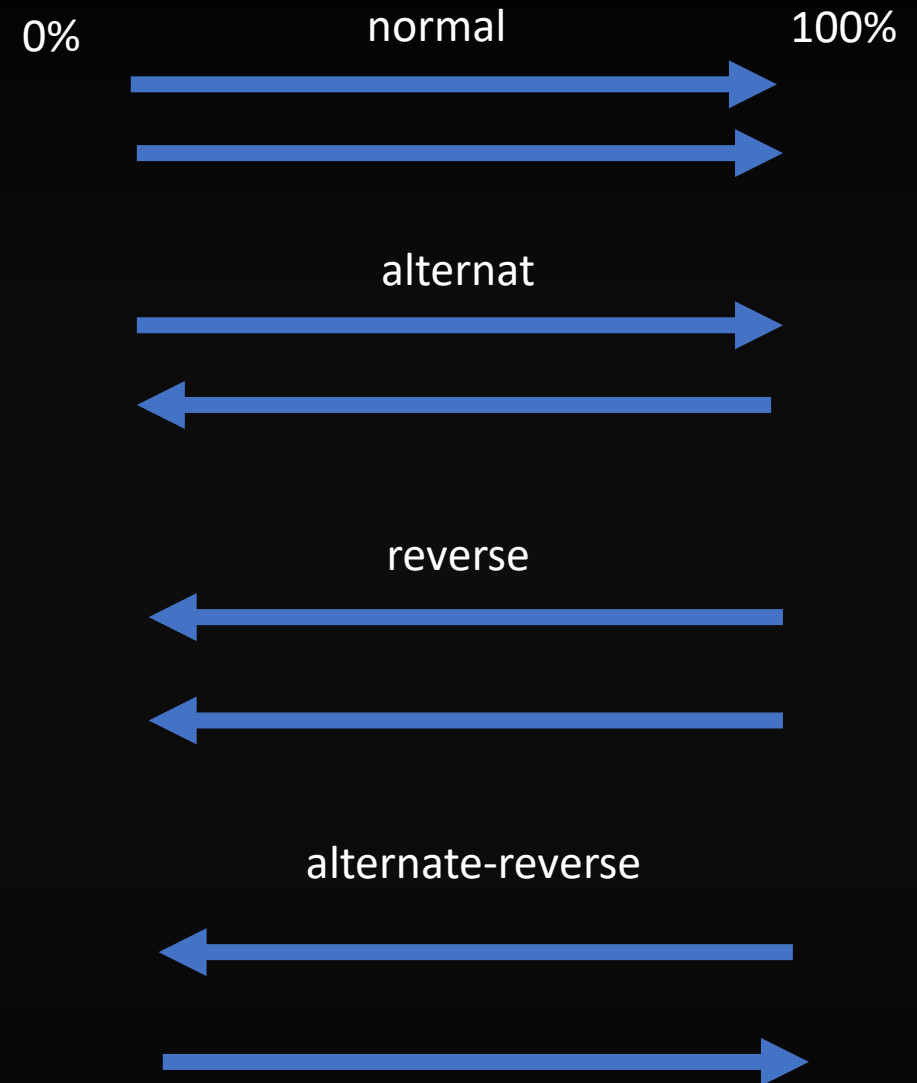
`animation-direction: reverse;`

Określa sposób wykonania animacji.

Domyślnie każda animacja od 0 do 100% (**normal**). Jeśli chcemy by w obu kierunkach to **alternate** (ale powrót to kolejne wykonanie)

`animation: superAnimacja 10s 2s alternate`

Jeśli nie użyjemy tej właściwości to domyślna wartość to **normal**;



animation-iteration-count

animation-iteration-count: 100;

Określa ile razy ma być wykonana animacja. Możemy wskazać konkretną wartość 1, 1000 a nawet ułamkową 1.2; Najczęściej odtwarzamy animację nieskończoną, która jest określona przez wartość **infinite**.

animation: superAnimacja 10s 2s alternate infinite

Jeśli nie użyjemy tej właściwości to domyślna jest 1 czyli animacja odtworzy się raz.

animation-fill-mode

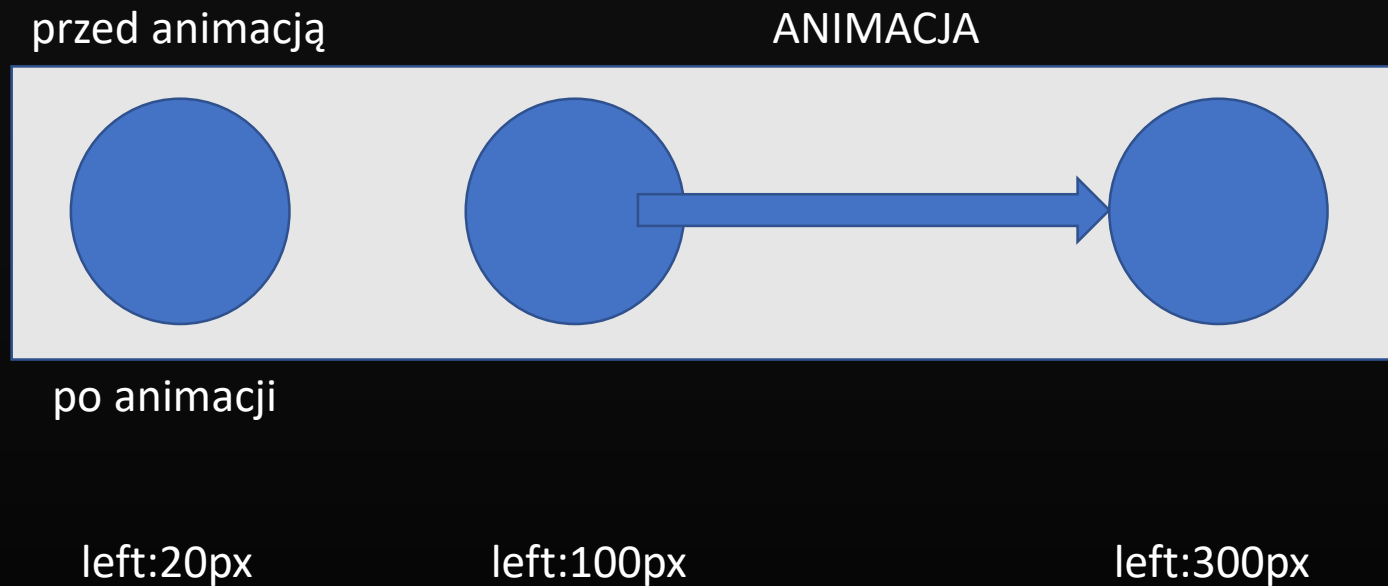
animation-fill-mode: both

Najtrudniejsza do zrozumienia :)

Ważne jeśli animacja nie uruchamia się od raz lub nie jest nieskończona.

animation-fill-mode

animation-fill-mode: none;



```
div {  
    position: absolute;  
    left:20px  
    top: 0px  
    animation: move 2s 2s  
}
```

```
@keyframes move {  
    0% { left:100px}  
    100% {left: 300px}  
}
```

Wartości z animacji nie
są wykorzystywane
przed ani po animacji

animation-fill-mode

animation-fill-mode: both;



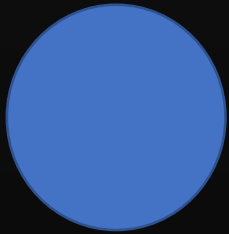
```
div {  
    position: absolute;  
    left:20px  
    top: 0px  
    animation: move 2s 2s both  
}  
  
@keyframes move {  
    0% { left:100px}  
    100% {left: 300px}  
}
```

Przed animacją jak i po animacji element przyjmuje wartości z animacji.

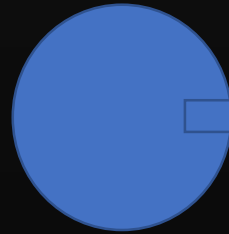
animation-fill-mode

animation-fill-mode: forwards;

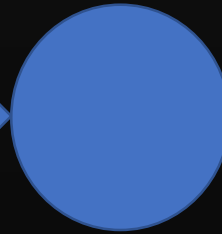
przed animacją



left:20px



left:100px



po animacji

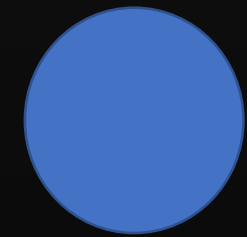
left:300px

```
div {  
    position: absolute;  
    left:20px  
    top: 0px  
    animation: move 2s 2s forwards  
}  
  
@keyframes move {  
    0% { left:100px}  
    100% {left: 300px}  
}
```

Po animacji element
zostaje w miejscu
zakończenia animacji.

animation-fill-mode

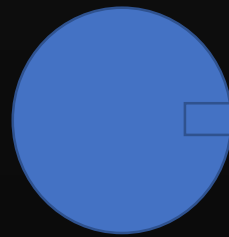
animation-fill-mode: backwards;



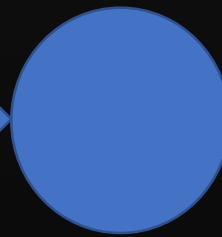
po animacji

left:20px

przed animacją



left:100px



left:300px

```
div {  
    position: absolute;  
    left:20px  
    top: 0px  
    animation: move 2s 2s backwards  
}  
  
@keyframes move {  
    0% { left:100px}  
    100% {left: 300px}  
}
```

Przed animacją element
jest w miejscu rozpoczęcia
animacji.

animation-fill-mode

animation-fill-mode: both

Najtrudniejsza do zrozumienia :)

animation: superAnimacja 10s 2s alternate Infinite both

Jeśli nie użyjemy tej właściwości to domyślna jest **none**, która przed rozpoczęciem animacji ustawia element taki jakby animacji nie było i to samo robi po animacji.

animation-timing-function

animation-timing-function: linear

można też kroku (steps), albo cubic-bezier, ease itd.

animation: superAnimacja 10s 2s alternate Infinite both linear

Domyślnie ease.

animation-play-state

animation-play-state: running

Można też ustawić paused

animation: superAnimacja 10s 2s alternate Infinite both linear
running

Domyślnie running. Paused zatrzymuje animacje na jakimś jej etapie (animacja się nie odtwarza a wartości są zamrożone).

Wiele animacja dla jednego obiektu.

```
p {  
  animation: wLewo 10s infinite, kolor 3s 10 2s;  
}
```

```
@keyframes kolor {  
}
```

```
@keyframes wLewo {  
}
```