

# Animation and effects cheat sheet

## Transform property

### Syntax

transform: transform function-values

### Example

```
1 .sample-class {  
2   transform: rotate(60deg);  
3 }
```

#### Keyword-value type: none

```
1 .sample-class {  
2   transform: none;  
3 }
```

#### Function-value type: matrix()

Variations: matrix(), matrix3d()

```
1 .sample-class {  
2   transform: matrix(1.0, 2.0, 3.0, 4.0, 5.0, 6.0);  
3 }
```

#### Function-value type: rotate(deg)

Variations: rotate(), rotate3d(), rotateX(), rotateY(), rotateZ()

```
1 .sample-class {  
2   transform: rotate3d(3,2,1, 100deg);  
3 }
```

Note: In rotate3d(), the respective values represent x, y, z co-ordinate and degree of rotations

#### Function-value type: translate(x,y)

Variations: translate(), translate3d(), translateX(), translateY(), translateZ()

```
1 .sample-class {  
2   transform: translate3d(10px, 20px, 30px);  
3 }
```

Note: In translate3d(), the respective values represent translation along the x, y, z co-ordinates

#### Function-value type: scale(factor)

Variations: scale(), scale3d(), scaleX(), scaleY(), scaleZ()

```
1 .sample-class {  
2   transform: scale3d(2, 1, 0.3);  
3 }
```

Note: In scale3d(), the respective values represent scaling times along the x, y, z co-ordinates

#### Function-value type: skew(deg, deg)

Variations: skew(), skewX(), skewY()

```
1 .sample-class {  
2   transform: skew(100deg);  
3 }
```

#### Global value types:

```
1 .sample-class {  
2   transform: inherit;  
3 }
```

```
1 .sample-class {  
2   transform: initial;  
3 }
```

```
1 .sample-class {  
2   transform: revert;  
3 }
```

```
1 .sample-class {  
2   transform: revert-layer;  
3 }
```

```
1 .sample-class {  
2   transform: unset;  
3 }
```

## Multiple transform over the same element

### Syntax

Transform can be applied for rotate(), scale() and translate() that can be listed together. Each of these properties can have their own values and the actions will give a combined effect.

### Example

```
1 .sample-class {  
2   transform: rotate(45deg) scale(1.5) translate(45px);  
3 }
```

Additional property under transform:transform-origin

Determines the anchor point for the centering of transform.

### Example

```
1 .sample-class {  
2   transform-origin: 10px 10px;  
3 }
```

```
1 .sample-class {  
2   transform-origin: right bottom;  
3 }
```

## Transition property

### Transition shorthand

Transition shorthand has four following sub-properties, each of which can also be individually defined.

- transition-property
- transition-duration
- transition-timing-function
- transition-delay

You have to list the values without naming them individually. Values skipped will be assigned their default values.

### Syntax

transition: property duration timing-function delay;

## Example

transition: margin-left 2s ease-in-out 0.5s;

## Animations and @keyframes

animation property:

### Syntax

animation: *name duration timing-function delay iteration-count direction fill-mode play-state*;

## Example

```
1 .sample-class {  
2   animation: none 2 ease 0.5 4 normal none running;  
3 }
```

The animation property is a shorthand for the sub-properties below:

```
1 animation-name  
2 animation-duration  
3 animation-timing-function  
4 animation-delay  
5 animation-iteration-count  
6 animation-direction  
7 animation-fill-mode  
8 animation-play-state
```

The values not mentioned are given default values.

Animation-name property is used to tie-in the @keyframes rule.

## @keyframes

### Syntax

```
1 @keyframes mymove {  
2   from {property: value}  
3   to {property: value }  
4 }
```

## Example

```
1 @keyframes animation-name {  
2   from {bottom: 0px;}  
3   to {bottom: 100px;}  
4 }
```

Percentage denotes the timing of the animation.

### Alternative syntax

```
1 @keyframes animation-name {  
2   /* declare actions here */  
3 }
```

## Example

```
1 @keyframes animation-name {  
2   0%,100%{  
3     background-color: blue;  
4   }  
5   50% {  
6     background-color: green;  
7   }  
8 }
```

## Multiple animations

Works the same as regular animation, multiple rules can be set.

```
1 #some-class{  
2   animation: animation-a 2s linear infinite alternate,  
3   animation-b 3s ease infinite alternate;  
4 }
```

✓ Completed

Go to next item



Like



Dislike



Report an issue