netlink
digital solutions

Animation

www.netlink.com

Animation

- The animation property in CSS can be used to animate many other CSS properties such as color, background-color, height, or width.

- Each animation needs to be defined with the @keyframes at rule which is then called with the animation property.

1. Transformation - The transform property applies a 2D or 3D transformation to an element. This property allows you to rotate, scale, move, skew, etc., elements
2.Transitions – Performing the transformations smoothly. CSS transitions allows you to change property values smoothly, over a given duration.
3.Keyframes – Changing the animation (property, value, etc.) at a given time or state.

The properties that come with the CSS animations can be divided into three major categories:

1. **Transformation** – Transforming the dimensions, rescaling the objects, moving them from point A to B, etc.

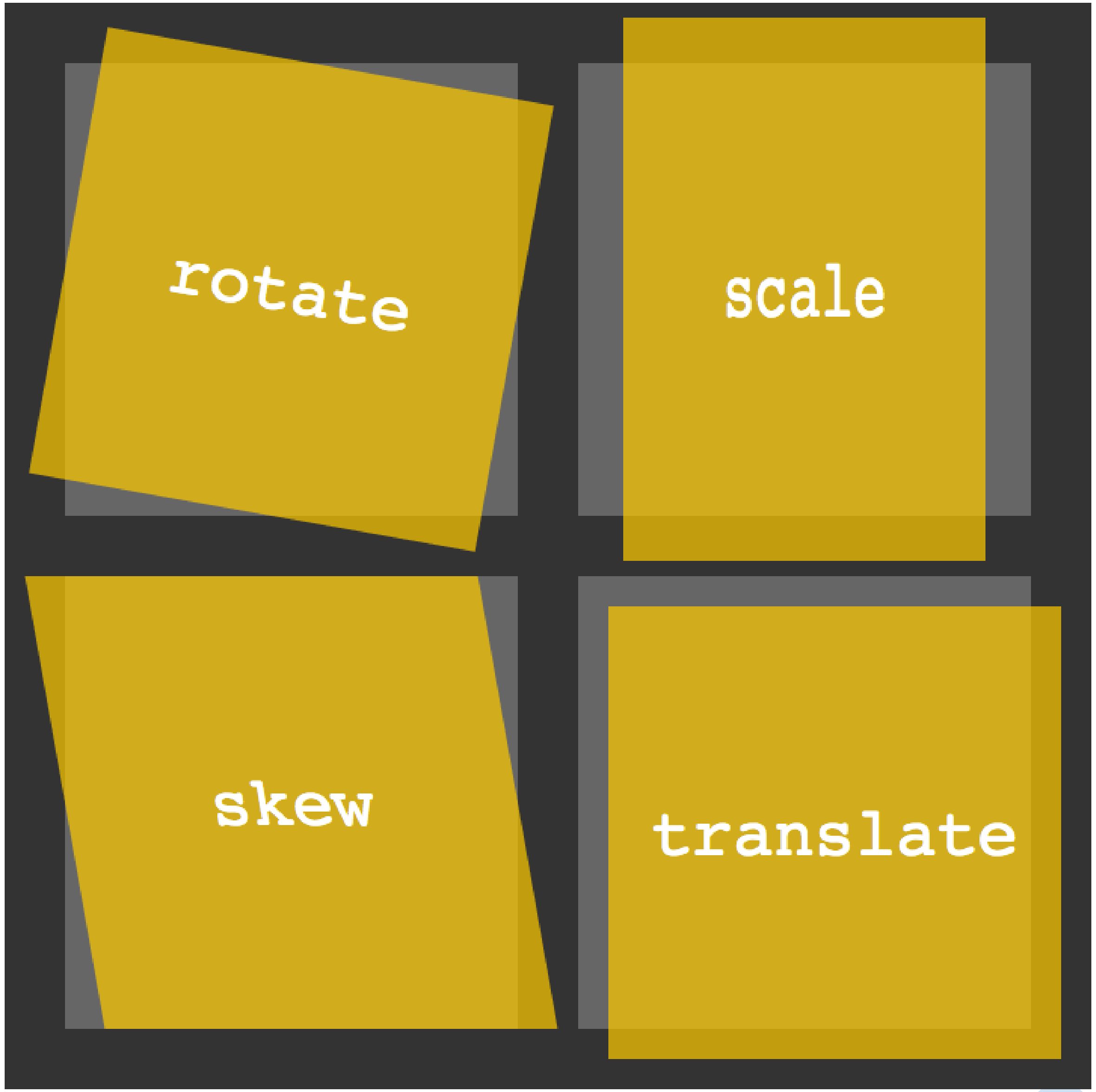
- The transform property applies a 2D or 3D transformation to an element. This property allows you to rotate, scale, move, skew, etc., elements.

translate: moves the element along up to 3 axis (x,y and z)

rotate: moves the element around a central point

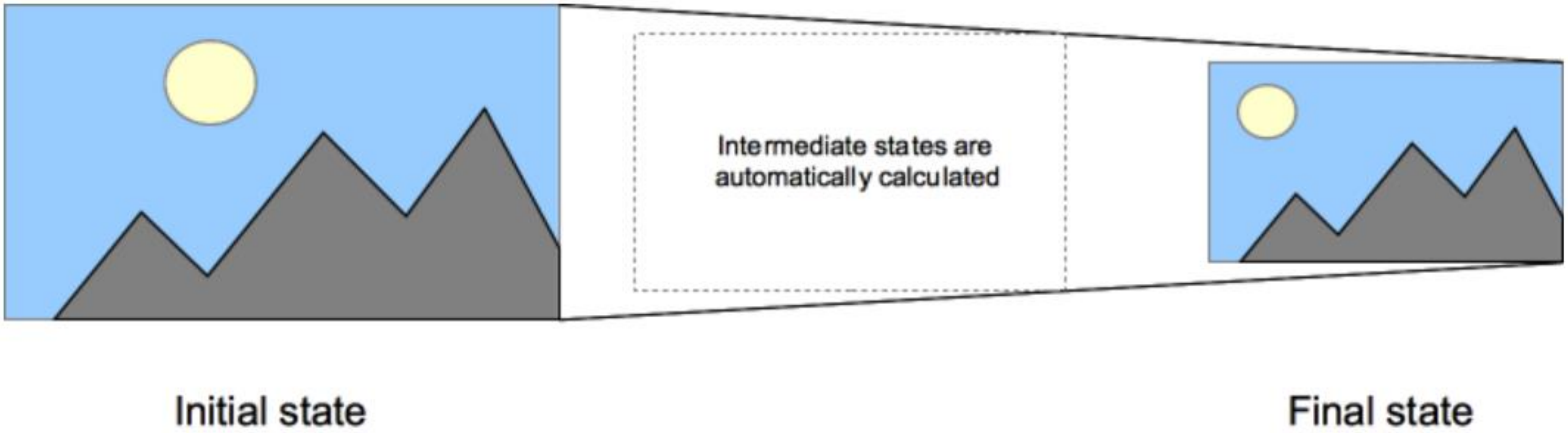
scale: resizes the element

skew: distorts the element


rotate
scale
skew
translate

2. Transition

- CSS transitions provide a way to control animation speed when changing CSS properties.
- Instead of having property changes take effect immediately, you can cause the changes in a property to take place over a period of time.
- For example, if you change the color of an element from white to black, usually the change is instantaneous.
- With CSS transitions enabled, changes occur at time intervals that follow an acceleration curve, all of which can be customized.


Intermediate states are automatically calculated
Initial state
Final state

Simple example

This example performs a four-second font size transition with a two-second delay between the time the user mouses over the element and the beginning of the animation effect:

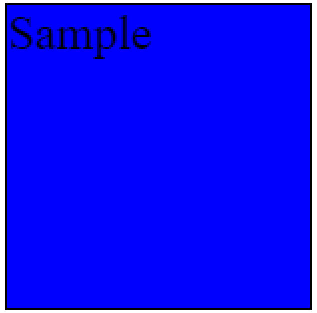
```
#delay {
  font-size: 14px;
  transition-property: font-size;
  transition-duration: 4s;
  transition-delay: 2s;
}

#delay:hover {
  font-size: 36px;
}
```

Multiple animated properties example

```
.box {
   border-style: solid;
   border-width: 1px;
   display: block;
   width: 100px;
   height: 100px;
   background-color: #0000FF;
   transition: width 2s, height 2s, background-color 2s, transform 2s;
}

.box:hover {
   background-color: #FFCCCC;
   width: 200px;
   height: 200px;
   transform: rotate(180deg);
}
```



Sample

Animation

3. The @keyframes Rule

When you specify CSS styles inside the @keyframes rule, the animation will gradually change from the current style to the new style at certain times.
To get an animation to work, you must bind the animation to an element.

Animation

The sub-properties of the animation property are:

❑ **animation-name**

Specifies the name of the @keyframes at-rule describing the animation's keyframes.

❑ **animation-duration**

Configures the length of time that an animation should take to complete one cycle.

❑ **animation-timing-function**

Configures the timing of the animation; that is, how the animation transitions through keyframes, by establishing acceleration curves.

❑ **animation-delay**

Configures the delay between the time the element is loaded and the beginning of the animation sequence.

Animation

❑ **animation-iteration-count**

Configures the number of times the animation should repeat; you can specify infinite to repeat the animation indefinitely.

❑ **animation-direction**

Configures whether or not the animation should alternate direction on each run through the sequence or reset to the start point and repeat itself.

❑ **animation-fill-mode**

Configures what values are applied by the animation before and after it is executing.

❑ **animation-play-state**

Lets you pause and resume the animation sequence.

Thank you