TRANSITIONS, TRANSFORMS, & ANIMATIONS with CSS3

COMP 126: Practical Web Design & Development for Everyone

transitions

allow property changes in CSS values to occur smoothly over a specified duration

animations

you can combine transitions and transforms to animate elements

transforms

allow elements styled with CSS to be moved or changed in two-dimensional or three-dimensional space

transitions 

simple nav transition

//codepen.io/tkjn/embed/RdOmMq/?height=265&theme-id=0&default-tab=css,result

anatomy of a transition

- **transition-property:** the property you want to transition. Note: not every property can be transitioned.
- **transition-duration:** in seconds or milliseconds: 4s or 4000ms (250-300ms is considered the "sweet spot").
- **transition-timing-function:** “cushioning" for the transition, defaults to ease; **optional.**
- **transition-delay:** the number of milli/seconds to delay the transition before firing it; **optional.**

Shorthand syntax:

```
transition: [property] [duration] [timing-
            function][delay];
```

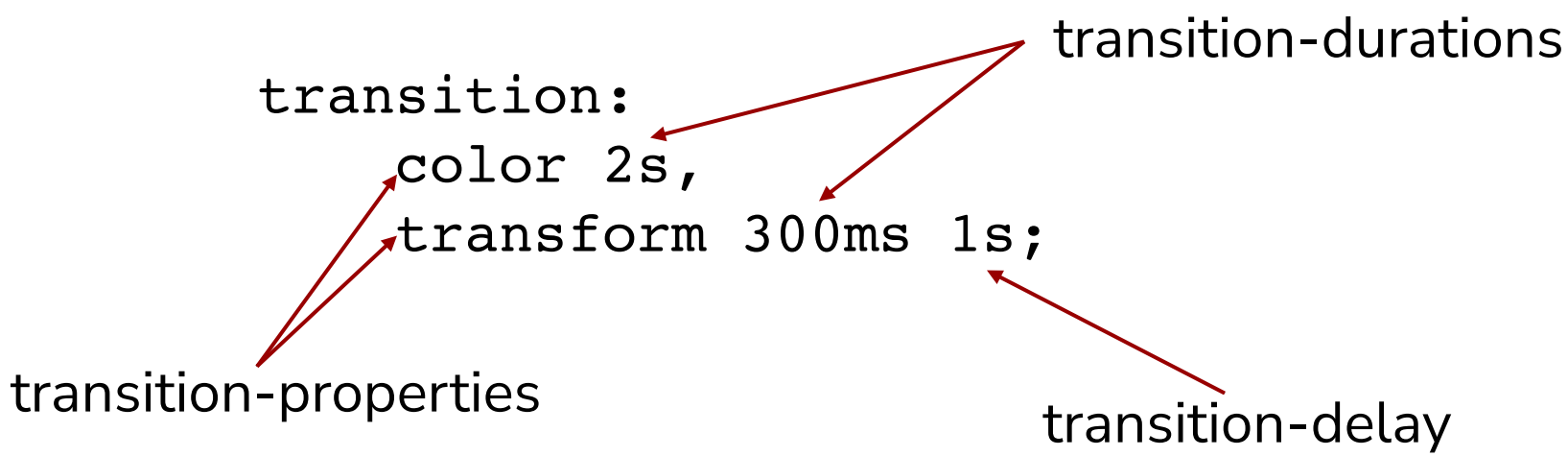
simple hover transition

//codepen.io/tkjn/embed/xQGZzK/?height=265&theme-id=0&default-tab=css,result

transitioning multiple

```
transition-property: color, transform;
transition-duration: 2s, 300ms;
transition-delay: 0, 1s;
```

or, in shorthand:



transforms 

transforms

...allow you to change the *shape and positioning of an element* without interfering with the document flow (the other elements on the page); can be coupled with transitions for an animation-style effect

translate (move)

rotate/scale (size)

skew (distortion/squish)

2D transforms

- horizontal (x-axis)
- vertical (y-axis)

3D transforms

- horizontal (x-axis)
- vertical (y-axis)
- depth (z-axis)

simple transform

//codepen.io/tkjn/embed/aQONVB/?height=265&theme-id=0&default-tab=css,result

more transitions & transforms

//codepen.io/tkjn/embed/MzwaEo/?height=265&theme-id=0&default-tab=css,result

transform-origin

changes the position of the origin of transformation from its default origin (i.e., its own center and/or the boundaries of the parent element); values, examples, and more demos here

transform-origin

Just a cool little CSSTricks demo to show you how different transform-origin values affect transformations

//codepen.io/tkjn/embed/mQVWxy/?height=265&theme-id=0&default-tab=css,resultundefined&editable=true


animations

animation property + @keyframes block

```
.animated-div {
    animation: black-to-white 1s linear 1;
}

@keyframes black-to-white {
    0% { background: black; }
    100% { background: white; }
}
```

the **animation property** binds a named animation to an element and specifies its duration

the **@keyframes block** names the animation and specifies what its properties are animating "from" (at 0%) and "to" (at 100%)

animation property/values

```
.animated-div {
    animation: black-to-white 1s linear 2s 2;
}
```

- **animation-name:** The name of the keyframe block you want to use. Can be anything.
- **animation-duration:** How long the animation takes to proceed from 0% complete to 100% complete.
- **animation-timing-function**: Just like transition-timing-function; defaults to ease.
- **animation-delay:** The number of seconds to delay the start of the animation from its trigger; **optional.**
- **animation-iteration-count:** The number of times you want the animation to proceed from 0% to 100%; "infinite" if you don't want to it stop. Defaults to 1; **optional.**

the @keyframes block

...allows you to define a group of properties and values to animate, rather than simply changing one or two values (as in transitions)

start (0%): white text on black

the name I chose for this animation (black-to-white)

```
@keyframes black-to-white {
  0% /* or from */ {
    background: black;
    color: white;
  }
  100% /* or to */ {
    background: white;
    color: black;
  }
}
```

end (100%): black text on white

you can control the animation in stages with multiple percentages

```
@keyframes black-to-white
  0% { color: black; }
  50% { color: grey; }
  100% { color: white; }
}
```

timing functions

linear, ease, ease-in, ease-out, ease-in-out (just like with transitions)

- **linear**: The animation has the same speed from start to end.
- **ease**: Default value. The animation has a slow start, then fast, before it ends slowly.
- **ease-in**: The animation has a slow start.
- **ease-out**: The animation has a slow end.
- **ease-in-out**: The animation has both a slow start and a slow end.

multi-step animation

//codepen.io/tkjn/embed/pQgRvX/?height=265&theme-id=0&default-tab=css,result

gradient with transition

https://codepen.io/tkjn/embed/yLobbay?default-tab=html%2Cresult

buttons: transition, transform, animation

https://codepen.io/tkjn/embed/eYEeNeX?default-tab=html%2Cresult

CSS animation in the wild

https://www.sliderrevolution.com/resources/css-animation-examples/

https://animate.style/