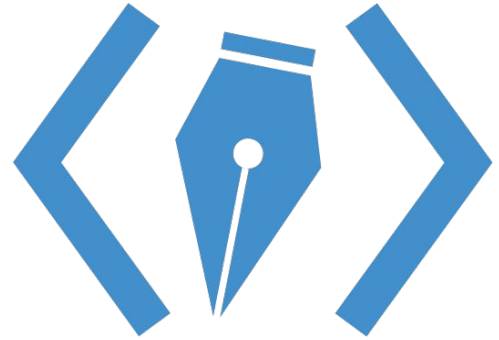


Week 6

CSS Animations



Announcements

Portal

Currently trying to fix the issue, but may experience crashes. We appreciate your patience!

Lab Section this Thursday

If you're in **Bianca, Howie, Ajia, Julia, or Jay's** section, please go to someone else's lab this week!

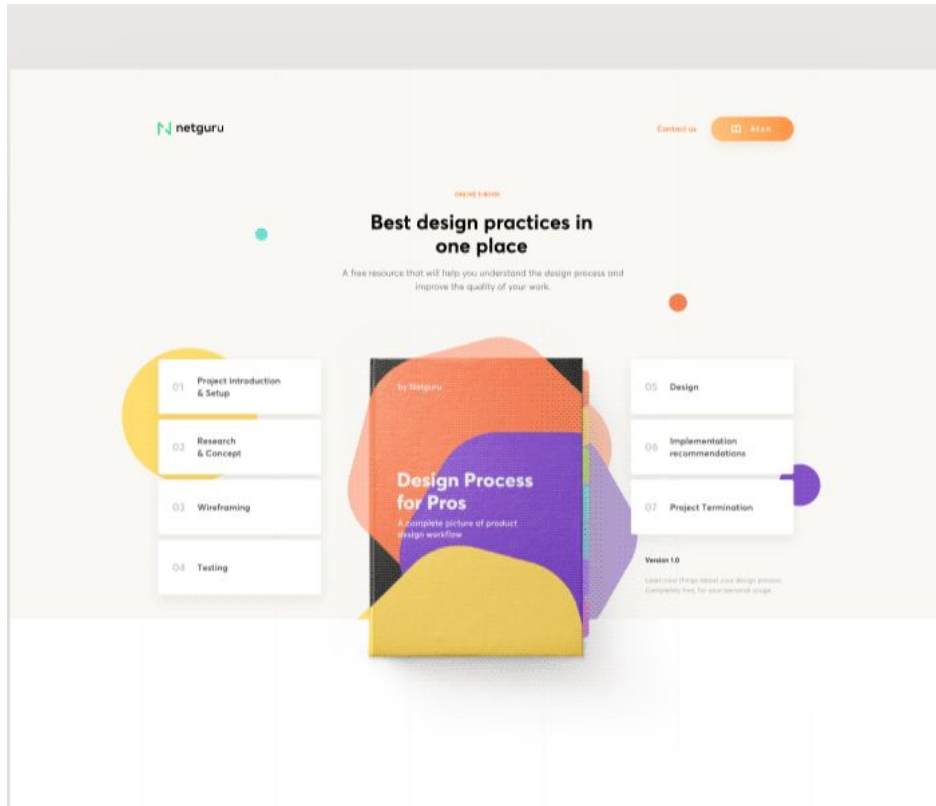
Midterm Project

The final project is due on **Tues, Oct 20th @ 11:59pm**.

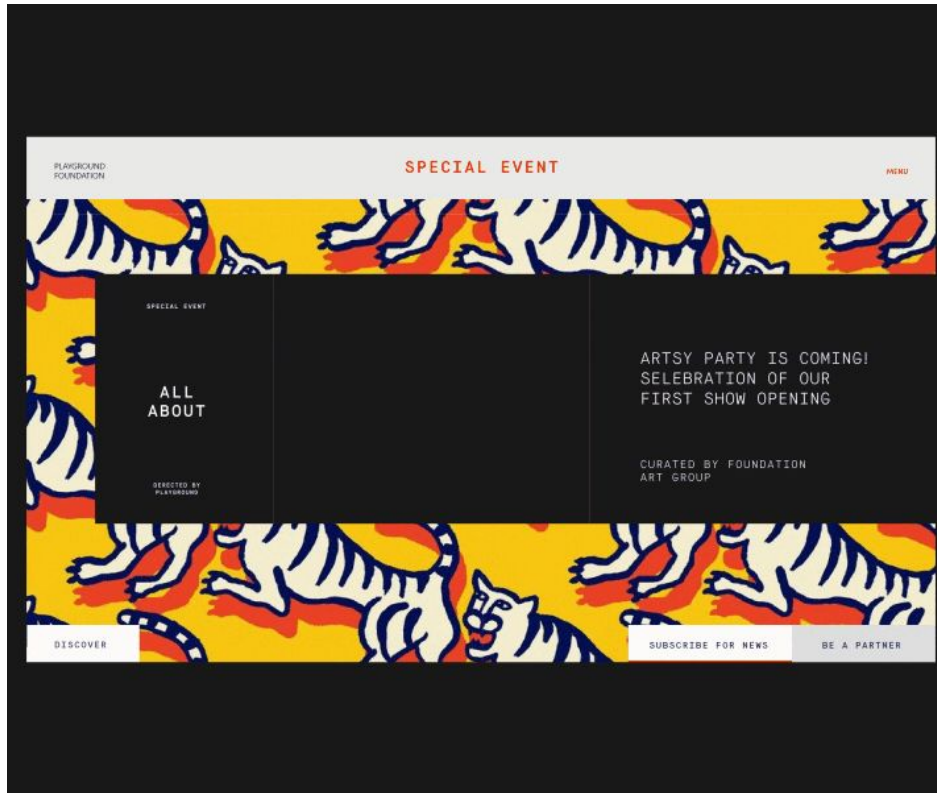
Submission (due 10/20) : BOTH ON PORTAL AND wdd.io/go/project-submit

Come to office hours via Piazza or wdd.io/go/OH

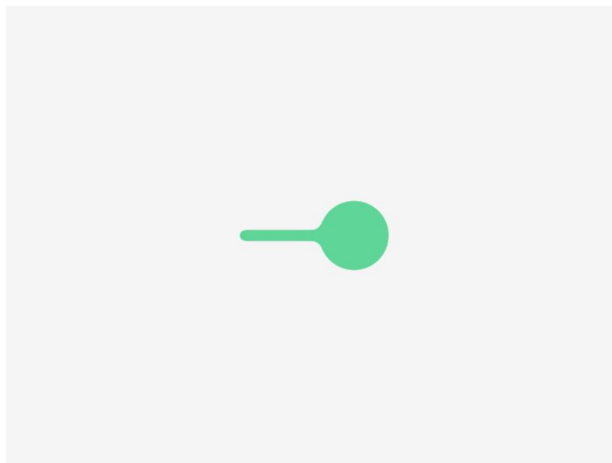
Give us anonymous feedback at wdd.io/go/feedback



With animations,
we can create
really nice,
dynamic websites.



Effectively allowing
to **express** our
brands or
identities in ways
we haven't before.



Animation - a change of state over time

What is our starting state? What is our ending state?

Most of which, can be done in CSS.

@keyframes

- Format: @keyframes **name** {**rules**}
- Rules: what to do during animation
- Defining animation to use **later**

```
@keyframes name {  
  rules  
}
```

Writing rules

- Using from and to
- Using percentages [0 - 100]

```
@keyframes fadeIn {  
  from { opacity: 0; }  
  to { opacity: 1; }  
}
```


Writing rules

- Using from and to
- **Using percentages [0 - 100]**

```
@keyframes fadeInOut {  
  0% { opacity: 0; }  
  50% { opacity: 1; }  
  100% { opacity: 0; }  
}
```

Writing rules: another example

- Using from and to
- **Using percentages [0 - 100]**

```
@keyframes upDown {  
  0% { top: 100px; }  
  50% { top: 0px; }  
  100% { top: 100px; }  
}
```

9:41



Let's Dive Into Design.

Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Curabitur varius nibh
non metus pharetra, at consectetur arcu.

Next →

Start: 0%
End: 100%

9:41



Let's Dive Into Design.

Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Curabitur varius nibh
non metus pharetra, at consectetur arcu.

Next →

Mid: 50%

9:41



Let's Dive Into Design.

Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Curabitur varius nibh
non metus pharetra, at consectetur arcu.

Next →

Notice the differences -- don't worry about what's in between

Application

Using your keyframe

- Put it in the element of choice
 - Use the name you chose
- Set the animation duration
- Add optional adjustments

```
@keyframes upDown {  
  0% { top: 100px; }  
  50% { top: 0px; }  
  100% { top: 100px; }  
}
```



```
#potato-head {  
  animation-name: upDown;  
  animation-duration: 0.5s;  
}
```

Mandatory properties

- Name
- Duration

```
#potato-head {  
  animation-name: upDown  
  animation-duration: 0.5s;  
}
```

Must include these or else your animation won't work!

Optional adjustments?

- **Speed curve**
- Delay
- Play #
- Direction

```
#potato-head {  
  animation-name: upDown  
  animation-duration: 0.5s;  
  animation-timing-function: ease-in;  
  animation-delay: 0s;  
  animation-iteration-count: 2;  
  animation-direction: reverse;  
}
```



Easing Functions

Optional adjustments?

- Speed curve
- **Delay**
- Play #
- Direction

```
#potato-head {  
  animation-name: upDown  
  animation-duration: 0.5s;  
  animation-timing-function: ease-in;  
  animation-delay: 0s;  
  animation-iteration-count: 2;  
  animation-direction: reverse;  
}
```

Optional adjustments?

- Speed curve
- Delay
- **Play #**
- Direction

```
#potato-head {  
  animation-name: upDown  
  animation-duration: 0.5s;  
  animation-timing-function: ease-in;  
  animation-delay: 0s;  
  animation-iteration-count: 2;  
  animation-direction: reverse;  
}
```

Optional adjustments?

- Speed curve
- Delay
- Play #
- **Direction**

```
#potato-head {  
  animation-name: upDown  
  animation-duration: 0.5s;  
  animation-timing-function: ease-in;  
  animation-delay: 0s;  
  animation-iteration-count: 2;  
  animation-direction: reverse;  
}
```

Animation shorthand

In this order:

- Name
- Duration
- Speed curve
- Delay
- Play #
- Direction

```
#potato-head {  
    animation: upDown 0.5s ease-in 0s 2 reverse;  
}
```

Stop in final state

Use special property to stop animation in a specific final state

```
#potato-head {  
  animation: upDown 0.5s ease-in 0s 2 reverse;  
  animation-fill-mode: forwards;  
}
```

Transitions: Pseudo-selectors

Responding to a user's action

Use transition property to determine time, speed curve, and pseudo-selector to determine states

Example

```
div:hover { rules }
```

Animations: Keyframes

Continuous and automatically instigated animations

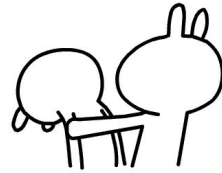
Does not depend on user states, more control over animation and timeframes

Example

```
div { animation: name }  
@keyframe name { rules }
```

Demo

wdd.io/go/keyframe-demo



Questions?