BEGINNING iOS ANIMATIONS

HANDS-ON CHALLENGES

Beginning iOS Animations

Catie & Jessy Catterwaul

Table of Contents: Overview

Table of Contents: Extended

Challenge 6: Beginning View Animations

By Catie & Jessy Catterwaul

Help! We want to make sandcastles, not snowmen.

In the demo, we mentioned that you'd be helping us manage the excess of snow. Get out your animation shovel and let's get started!

There are only two moving parts this time, and no helper views required.

Your goal is to fade the `snowView` in or out depending on the state of a flight's `showWeatherEffects` property.

Below the animation call you just wrote in the demo, at the bottom of `fade()`, add a new animation that includes `delay` and `options`:

```
    UIView.animate(
      withDuration: 1.0,
      delay: 0.0,
      options: [.curveEaseOut],
      animations: {

      },
      completion: nil
    )
```

You'll let this animation run longer than the one managing the background view. When combining animations, it's best if they don't all stop and start at the same time. This will allow a nice overlap of effects.

Now to actually animate the property, add this to the `animations` closure:

```
    self.snowView.alpha = showEffects ? 1.0 : 0.0
```

This will fade `snowView` in when the effects should be on, and fade it out when they should be off.

Build and run to see your handiwork!

Cleanup

Now you're going to perform a little cleanup. Notice when you first build and run the app, that the background image is animating. You don't really want that at first launch. `changeFlight` is already equipped to fix this. There is an unused `animated` bool being passed in to help you out.

Down in `changeFlight()` wrap the call to `fade` in an if statement:

```
if animated {
  fade(
    toImage: UIImage(named: data.weatherImageName)!,
    showEffects: data.showWeatherEffects
  )
} else {

}
```

If `animated` is true, your animation will be called. What if it's false?

There should still be a commented out line of code just above the `if` statement. Uncomment it, cut it, and add it to the `else` clause you just created:

```
    bgImageView.image = UIImage(named: data.weatherImageName)
```

Now when the app first launches the background image will be set without animation.

Give `snowView` a similar treatment. Add this to the `else` clause as well:

```
    snowView.isHidden = !data.showWeatherEffects
```

As you continue to add animations to this project, you'll follow a similar pattern inside of `changeFlight`. Each time you add an animation call to `if`, check to see if you should be moving an assignment to the `else` clause as well.

Build and run once more to see a animationless start, and beautiful background and snow fade animations thereafter!