Intermediate iOS Animation

Hands-on Challenges

Intermediate iOS Animation Hands-On Challenges

Challenge A: More animations

I hope you enjoyed that first video in the series! You didn't cover that much but you did learn how to create a layer animation and start it off – and you are going to use that knowledge in each of the videos that follow to create more and more complex animations.

In this video's challenge you are going to create a couple more animations to exercise what you've learned.

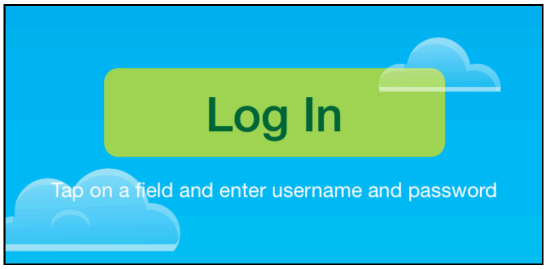
Let's get started.

Open **ViewController.swift** and find the `animateInfo()` method. It creates an instructional label and adds it to the view controller's view. What you are going to do is add animations to that label.

First add at the end of viewDidLayoutSubviews():

```
animateInfo()
```

Run the project and you will see the label show up below the Log In button:

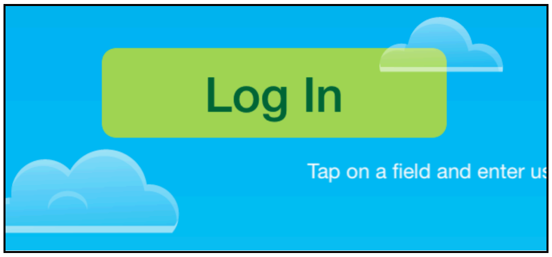


Now let's add animations to the label. Add an animation to fly the label from the right edge of the screen much like you did for the text fields:

```
let flyLeft = CABasicAnimation(keyPath: "position.x")
flyLeft.fromValue = info.layer.position.x + view.frame.size.width
flyLeft.toValue = info.layer.position.x
flyLeft.duration = 10.0
info.layer.addAnimation(flyLeft, forKey: nil)
```

You define an animation on the `position.x key` path of the layer, then you configure it to start from outside the screen and finish at its original position in the center of the view. You set a `10` seconds duration and you're off to the races!

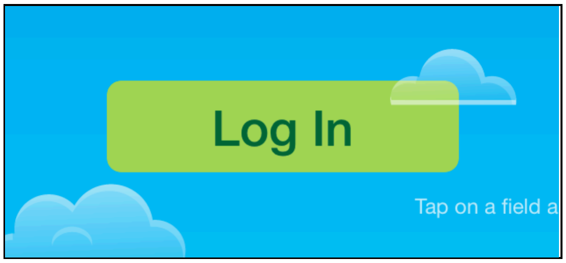
Run the project to check out the animation:



Now why not add one more animation to the same label? No reason not to, really. You can add as many animations you like to the same layer so just go ahead and add a fade in animation:

```
let fadeIn = CABasicAnimation(keyPath: "opacity")
fadeIn.fromValue = 0.0
fadeIn.toValue = 1.0
fadeIn.duration = 5.0
info.layer.addAnimation(fadeIn, forKey: nil)
```

The second animation animates the `opacity` of the label’s layer from `0` to `1` over a `5` second interval. You add it like usual to the label and now both animations will run at the same time:


Log In
Tap on a field a