

Spinner Animation in React - How to use Framer Motion to create a custom loading spinner animation

Written by Seth Corker on Benevolent Bytes

What we're making

We're creating a common spinner animation you've probably seen many times before. This is easy to create in CSS and we'll do just that but for the animation, we're going to make use of Framer Motion. With this technique, we could replace the component that is spinning and/or dynamically update it as it spins.

How to achieve the effect

I've made a quick video tutorial going through the process, it's about 3m 30s.

A 3 minute 30 second video tutorial showing how to create a circle spinner in Framer Motion and React

Because the animation is very simple, we are just making use of two props on the `motion` component and some careful setup.

Making the loop

The core props we need to change is `animate`, the animation we want to play (in our case we'll use `rotate: 360`). This will spin our circle exactly once so the next step is to provide a `transition` prop. We'll provide an object with 3 values, like so:

```
const spinTransition = {
  loop: Infinity,
  ease: "linear",
  duration: 1,
}
```

The transition object

1. `loop` is important to ensure the animation plays continuously.
2. `ease` is set to linear, this is important because by default, the animation will have easing which creates a completely different effect where it's slow to start and slows down again as it completes the rotation.
3. `duration` at 1 second gives a smooth and consistent rotation that isn't too fast, again, the default is very short which makes for a hectic and stressful loading spinner.

```
<motion.span
  style={circleStyle}
  animate={{ rotate: 360 }}
  transition={spinTransition}
/>
```

Connecting everything together That's all there is to it.

Where do I go from here?

To take this effect further, you could introduce dynamic values. Colour or size changes can easily be achieved by adjusting the `animate` prop. The transition object can also be fine tuned by passing in different values to the exact parameters you want to change.

```
const spinTransitionFineTuned = {
  rotate: {
    loop: Infinity,
    ease: "linear",
    duration: 1,
  },
  scale: {
    duration: 0.5,
  },
}
```

Fine tuning animations If we just provide `loop`, `ease` and `duration`, it affects every property animation.

Resources

- To see the code, take a look at the [GitHub repo](#).
- Check out my playlist of video tutorials covering animation in Framer Motion
- Take a look at the official Framer Motion documentation