References:

- React Animated:
  https://facebook.github.io/react-native/docs/animated
- List of transform properties for a browser:
  https://www.w3schools.com/cssref/css3_pr_transform.asp
- List of animation easings:
  https://easings.net
- Which easings we can use:
  https://facebook.github.io/react-native/docs/easing

You will need:

- **"git" installed on your computer.**
  - **Mac: git --version**
  - **PC: https://git-scm.com/download/win**
  - **Linux debian: sudo apt install git-all**
  - **Linux fedora: sudo yum install git**

Check out the repo we will be using:

```
git clone https://github.com/jon617/react-animation-demo.git
```

***if you want to cheat, download the finished code:

```
git clone https://github.com/jon617/react-animation-demo-finished.git
```

```
cd react-animation-demo
npm install

npm install animated
npm start
```

File structure of our little demo:

- App.js - this is our basic page that shows a logo. Let's animate that logo

First, let's make a button on the App.js file:

```
<button type="button">
  click me to animate
</button>
```

Add state to App.js. This gives us a way to trigger the animation start and end:

```
this.state = {
  doAnimation: false,
}
```

On constructor of App.js

```
this.spinValue = new Animated.Value( 0 );
```

This initializes the spin value which internally goes from 0 to 1.

Create functions in the App.js:

```
doSpin = () => {
  this.spinValue.setValue( 0 );
  Animated.timing(
    this.spinValue,
    {
      toValue: 1,  // go from 0 to 1
      duration: 1000,  // milliseconds, so 1000 = 1second
      easing: Easing.inOut(),
      useNativeDriver: true,
    }
  ).start( )
}
```

In App.js, render() section, create the spin variable:

```
  const spin = this.spinValue.interpolate({
    inputRange: [ 0, 1 ],
    outputRange: [ '0deg', '360deg' ]
  });
```

Change the <img> tag to <Animated.img>

In the <Animated.img> section, edit the style= and add:

```
style={{
    Transform: [{
        rotate: spin
    }]
}}
```

Now, create a script to set the doAnimation variable in the state, and create a function to start the animation:

```
startOrEndSpin = () => {
  this.setState({ doAnimation: true });
  this.doSpin();
}
```

And, set the onClick in the Button to run this.

Next, create the onSpinCompletion, and add to doSpin in the .start(), so it runs after each animation completes

We can change the duration, easing, etc.

Easing.inOut

```
import React, { Component } from 'react';
import './App.css';
import * as Animated from 'animated/lib/targets/react-dom';
import Easing from 'animated/lib/Easing';

class App extends Component {
  constructor( props ) {
    super( props );
    this.state = {
      doAnimation: false,
    }
    this.spinValue = new Animated.Value( 0 );
  }

  startOrEndSpin = () => {
    if ( this.state.doAnimation ) {
      this.setState({ doAnimation: false });
    } else {
      this.setState({ doAnimation: true });
      this.doSpin();
    }
  }

  doSpin = () => {
    this.spinValue.setValue( 0 );
    Animated.timing(
      this.spinValue,
      {
        toValue: 1,  // go from 0 to 1
        duration: 1000,  // milliseconds, so 1000 = 1second
        easing: Easing.inOut( Easing.quad ),
        useNativeDriver: true,
      }
    ).start( this.onSpinCompletion )
  }
  onSpinCompletion = () => {
    if ( this.state.doAnimation ) {
      this.doSpin();
    }
```

```
  }

  render() {

    const spin = this.spinValue.interpolate({
      inputRange: [ 0, 1 ],
      outputRange: [ '0deg', '360deg' ]
    });

    return (
      <div className="App">
        <div style={{ marginTop: 35 }}>
          <Animated.img
            src="a-logo.jpg"
            alt=""
            width={ 200 }
            height={ 200 }
            style={{
              transform: [{
                rotate: spin
              }]
            }}
          />
        </div>

        <br /><br /><br /><br />
        <button
          type="button"
          onClick={ this.startOrEndSpin }
        >
          click me to animate
        </button>
      </div>
    );
  }
}

export default App;
```