



1. Crie uma pasta dentro de **app/pages** chamada **Animations**
2. Dentro da pasta **Animation** crie os seguintes arquivos
  - a. index.js
  - b. WithoutLayout.js
  - c. Layout.js

## Dentro de app/pages/Animations/WithoutLayout.js:



```
import React, { Component } from 'react';
import {
  StyleSheet,
  Text,
  View,
  Button
} from 'react-native';
```

# Dentro de app/pages/Animations/WithoutLayout.js:



```
export default class WithoutLayout extends Component {
  static navigationOptions = ({ navigation }) => ({
    header: null,
    tabBarLabel: 'Sem Layout'
  });

  constructor() {
    super();
    this.state = {
      styleName: 'default'
    }
  }

  changeState(styleName) {
    this.setState({ styleName });
  }

  render() {
    ...
  }
}
```

## Dentro de app/pages/Animations/WithoutLayout.js, no método render:

```
...

render() {
  const { navigate, goBack } = this.props.navigation;

  return (
    <View style={styles.container}>
      <View style={styles[this.state.styleName]} />

      <View style={styles.btnContainer}>
        <Button
          title="alterar estado"
          onPress={() => this.changeState('active')}
        />
      </View>

      <View style={styles.btnContainer}>
        <Button
          title="voltar ao estado default"
          onPress={() => this.changeState('default')}
        />
      </View>
    </View>
  );
}

...
```

# Dentro de app/pages/Animations/WithoutLayout.js:

---

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#F5FCFF',
    justifyContent: 'center',
    alignItems: 'center'
  },
  default: {
    backgroundColor: 'blue',
    width: 100,
    height: 100
  },
  active: {
    backgroundColor: 'blue',
    width: 200,
    height: 200
  },
  btnContainer: {
    marginTop: 10
  }
});
```



1. Faça uma cópia do arquivo **WithoutLayout.js** e renomeie a cópia para **Layout.js**
2. Renomeie a classe **WithoutLayout** para **Layout**

## Dentro de app/pages/Animations/Layout.js:



```
import React, { Component } from 'react';
import {
  StyleSheet,
  Text,
  View,
  Button,
  UIManager, // <-- novo import
  LayoutAnimation // <-- novo import
} from 'react-native';
```

Dentro de app/pages/Animations/Layout.js, no constructor:



```
constructor() {  
  super();  
  UIManager.setLayoutAnimationEnabledExperimental  
  && UIManager.setLayoutAnimationEnabledExperimental(true);  
  
  this.state = {  
    styleName: 'default'  
  }  
}
```



Dentro de app/pages/Animations/Layout.js, no método changeState:



```
changeState(styleName) {  
  LayoutAnimation.configureNext(LayoutAnimation.Presets.spring);  
  this.setState({ styleName });  
}
```

## Dentro de app/pages/Animations/index.js:



```
import React, { Component } from 'react';
import {
  StyleSheet,
  Text,
  View,
  Button
} from 'react-native';
import { TabNavigator } from 'react-navigation';
import Layout from './Layout';
import WithoutLayout from './WithoutLayout';
```

## Dentro de app/pages/Animations/index.js:

```
const Animations = TabNavigator({
  WithoutLayout: { screen: WithoutLayout },
  Layout: { screen: Layout }
}, {
  tabBarPosition: 'bottom',
  animationEnabled: true,
  tabBarOptions: {
    activeTintColor: 'yellow',
  }
});

export default class AnimationsContainer extends Component {
  static navigationOptions = ({ navigation }) => ({
    drawerLabel: 'Animations',
    title: 'Animations',
    headerLeft:
      <View style={{ marginLeft: 10 }}>
        <Button title="menu" onPress={() => navigation.navigate('DrawerToggle')} />
      </View>
  });

  render() {
    return <Animations />;
  }
}
```

## Dentro de app/index.js:



```
...

import AnimationsContainer from './pages/Animations/';

const Drawer = DrawerNavigator({
  ...
  Animations: { screen: AnimationsContainer }
}, {
  contentComponent: props => <ScrollView><DrawerItems {...props} /></ScrollView>,
});

...
```



Veja o resultado!



1. Dentro da pasta **Animations** crie dois arquivos:
  - a. Animated.js
  - b. Ball.js

## Dentro de app/pages/Animations/Animated.js:



```
import React, { Component } from 'react';
import {
  StyleSheet,
  Text,
  View,
  Button
} from 'react-native';
import Ball from './Ball';
```

## Dentro de app/pages/Animations/Animated.js:



```
export default class Animated extends Component {
  static navigationOptions = ({ navigation }) => ({
    header: null,
    tabBarLabel: 'Animated'
  });

  render() {
    const { navigate, goBack } = this.props.navigation;

    return (
      <View style={styles.container}>
        <Ball />
      </View>
    );
  }
}
```



## Dentro de app/pages/Animations/Animated.js:




```
const styles = StyleSheet.create({  
  container: {  
    flex: 1,  
    backgroundColor: '#F5FCFF'  
  }  
});
```

## Dentro de app/pages/Animations/Ball.js:



```
import React, { PureComponent } from 'react';  
import { View, Animated, StyleSheet, TouchableWithoutFeedback } from 'react-native';
```

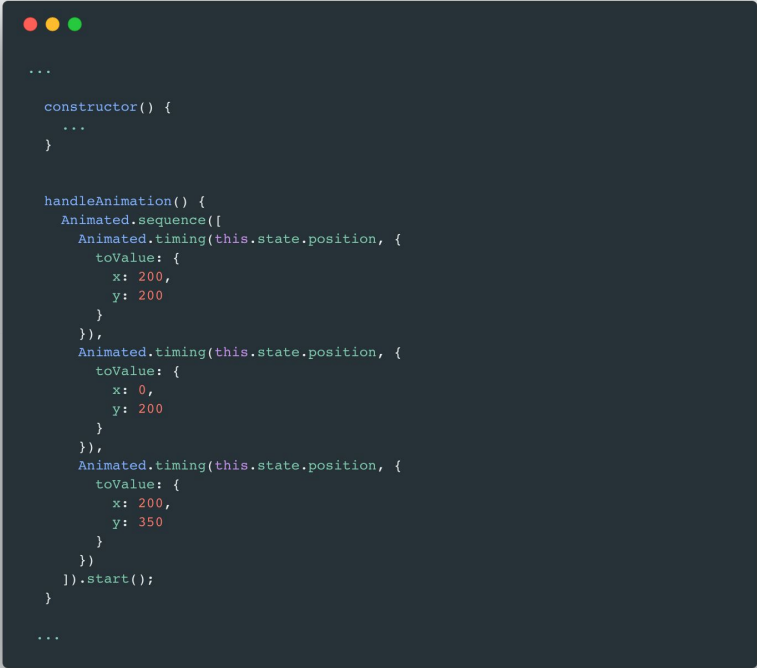
## Dentro de app/pages/Animations/Ball.js:



```
export default class Ball extends PureComponent {
  constructor() {
    super();
    this.state = {
      position: new Animated.ValueXY(0, 0)
    };
  }

  render() {
    return (
      <Animated.View style={this.state.position.getLayout()}>
        <TouchableWithoutFeedback onPress={() => this.handleAnimation()}>
          <View style={styles.ball} />
        </TouchableWithoutFeedback>
      </Animated.View>
    );
  }
}
```

## Dentro de app/pages/Animations/Ball.js, abaixo do constructor:




```
...  
  
constructor() {  
  ...  
}  
  
handleAnimation() {  
  Animated.sequence([  
    Animated.timing(this.state.position, {  
      toValue: {  
        x: 200,  
        y: 200  
      }  
    }),  
    Animated.timing(this.state.position, {  
      toValue: {  
        x: 0,  
        y: 200  
      }  
    }),  
    Animated.timing(this.state.position, {  
      toValue: {  
        x: 200,  
        y: 350  
      }  
    })  
  ]).start();  
}  
  
...
```

## Dentro de app/pages/Animations/Ball.js:



```
const styles = StyleSheet.create({  
  ball: {  
    width: 100,  
    height: 100,  
    backgroundColor: 'black',  
    borderRadius: 50  
  }  
});
```

## Dentro de app/pages/Animations/index.js, altere:



```
...  
  
import Animated from './Animated';  
  
const Animations = TabNavigator({  
  ...  
  Animated: { screen: Animated }  
}, {  
  tabBarPosition: 'bottom',  
  animationEnabled: true,  
  tabBarOptions: {  
    activeTintColor: 'yellow',  
  }  
});  
  
...
```



Veja o resultado!