**CHƯƠNG 3: Component trong react native.**

**Nhóm 3**

Props :

import React, { Component } from 'react';

import { Text, View } from 'react-native';

class Greeting extends Component {

  render() {

    return (

      <View style={{alignItems: 'center'}}>

        <Text>Hello {this.props.name}!</Text>

      </View>

    );

  }

}

export default class LotsOfGreetings extends Component {

  render() {

    return (

      <View style={{alignItems: 'center', top: 50}}>

        <Greeting name='Rexxar' />

        <Greeting name='Jaina' />

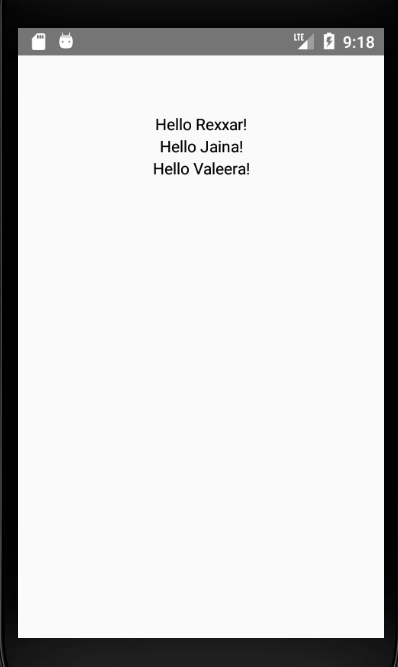
        <Greeting name='Valeera' />

      </View>

    );

  }

}



State:

import React, { Component } from 'react';

import { Text, View } from 'react-native';

class Blink extends Component {

  componentDidMount(){

    // Toggle the state every second

    setInterval(() => (

      this.setState(previousState => (

        { isShowingText: !previousState.isShowingText }

      ))

    ), 1000);

  }

  //state object

  state = { isShowingText: true };

  render() {

    if (!this.state.isShowingText) {

      return null;

    }

    return (

      <Text>{this.props.text}</Text>

    );

  }

}

export default class BlinkApp extends Component {

  render() {

    return (

      <View>

        <Blink text='I love to blink' />

        <Blink text='Yes blinking is so great' />

        <Blink text='Why did they ever take this out of HTML' />

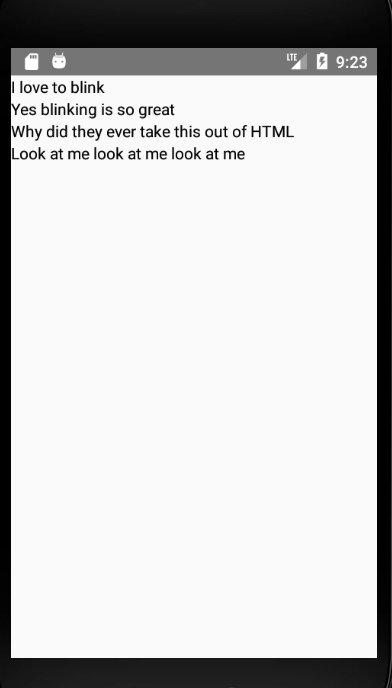
        <Blink text='Look at me look at me look at me' />

      </View>

    );

  }

}



Style trong react:

import React, { Component } from 'react';

import { StyleSheet, Text, View } from 'react-native';

const styles = StyleSheet.create({

  bigBlue: {

    color: 'blue',

    fontWeight: 'bold',

    fontSize: 30,

  },

  red: {

    color: 'red',

  },

});

export default class LotsOfStyles extends Component {

  render() {

    return (

      <View>

        <Text style={styles.red}>just red</Text>

        <Text style={styles.bigBlue}>just bigBlue</Text>

        <Text style={[styles.bigBlue, styles.red]}>bigBlue, then red</Text>

        <Text style={[styles.red, styles.bigBlue]}>red, then bigBlue</Text>

      </View>

    );

  }

}



Width và Height:

import React, { Component } from 'react';

import { View } from 'react-native';

export default class FixedDimensionsBasics extends Component {

  render() {

    return (

      <View>

        <View style={{width: 50, height: 50, backgroundColor: 'powderblue'}} />

        <View style={{width: 100, height: 100, backgroundColor: 'skyblue'}} />

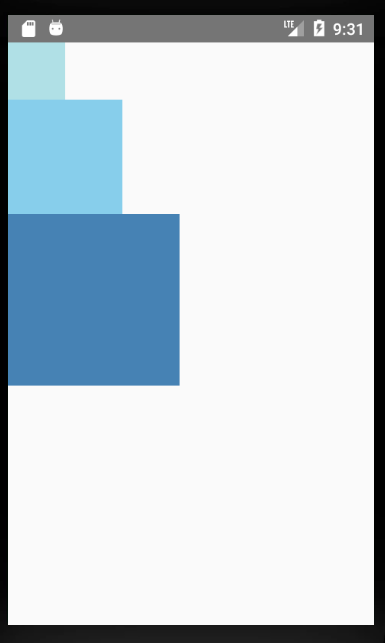
        <View style={{width: 150, height: 150, backgroundColor: 'steelblue'}} />

      </View>

    );

  }

}



Flex:

import React, { Component } from 'react';

import { View } from 'react-native';

export default class FlexDimensionsBasics extends Component {

  render() {

    return (

      // Try removing the `flex: 1` on the parent View.

      // The parent will not have dimensions, so the children can't expand.

      // What if you add `height: 300` instead of `flex: 1`?

      <View style={{flex: 1}}>

        <View style={{flex: 1, backgroundColor: 'powderblue'}} />

        <View style={{flex: 2, backgroundColor: 'skyblue'}} />

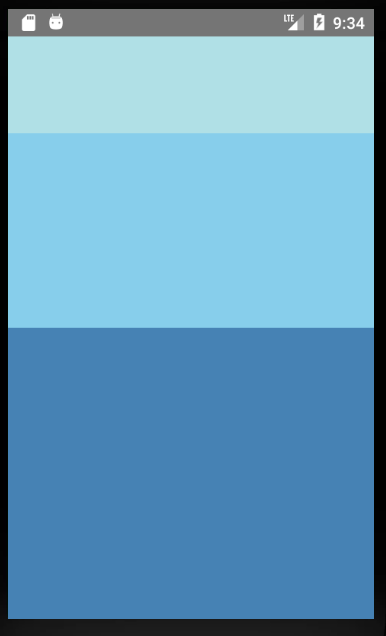
        <View style={{flex: 3, backgroundColor: 'steelblue'}} />

      </View>

    );

  }

}



FlexBox:

import React, { Component } from 'react';

import { View } from 'react-native';

export default class FlexDirectionBasics extends Component {

  render() {

    return (

      // Try setting `flexDirection` to `column`.

      //<View style={{flex: 1, flexDirection: 'row'}}>

      //<View style={{flex: 1, flexDirection: 'row-reverse'}}>

        <View style={{flex: 1, flexDirection: 'column'}}>

        <View style={{width: 50, height: 50, backgroundColor: 'powderblue'}} />

        <View style={{width: 50, height: 50, backgroundColor: 'skyblue'}} />

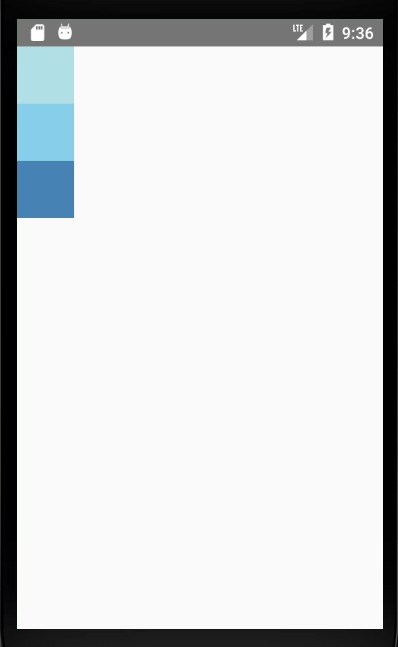
        <View style={{width: 50, height: 50, backgroundColor: 'steelblue'}} />

      </View>

    );

  }

};



Text Component:

import React, { Component } from 'react';

import { Text, StyleSheet } from 'react-native';

export default class TextInANest extends Component {

  constructor(props) {

    super(props);

    this.state = {

      titleText: "Bird's Nest",

      bodyText: 'This is not really a bird nest.'

    };

  }

  render() {

    return (

      <Text style={styles.baseText}>

        <Text style={styles.titleText} onPress={this.onPressTitle}>

          {this.state.titleText}{'\n'}{'\n'}

        </Text>

        <Text numberOfLines={5}>

          {this.state.bodyText}

        </Text>

      </Text>

    );

  }

}

const styles = StyleSheet.create({

  baseText: {

    fontFamily: 'Cochin',

  },

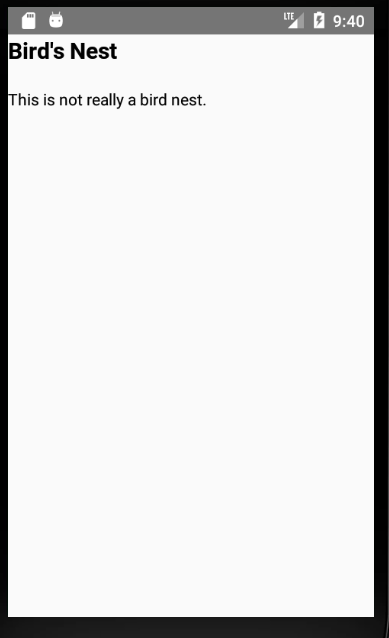
  titleText: {

    fontSize: 20,

    fontWeight: 'bold',

  },

});



Text input Component:

import React, { Component } from 'react';

import { Text, TextInput, View } from 'react-native';

export default class PizzaTranslator extends Component {

  constructor(props) {

    super(props);

    this.state = {text: ''};

  }

  render() {

    return (

      <View style={{padding: 10}}>

        <TextInput

          style={{height: 40}}

          placeholder="Type here to translate!"

          onChangeText={(text) => this.setState({text})}

          value={this.state.text}

        />

        <Text style={{padding: 10, fontSize: 42}}>

          {this.state.text.split(' ').map((word) => word && '🍕').join(' ')}

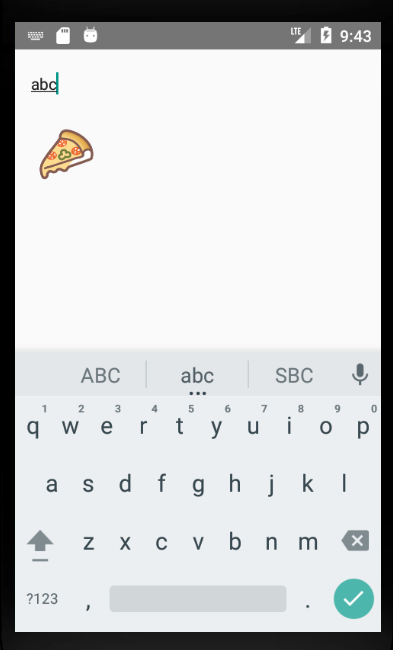
        </Text>

      </View>

    );

  }

}



Button Component:

import React, { Component } from 'react';

import { Text, TextInput, View, Button, Alert } from 'react-native';

export default class PizzaTranslator extends Component {

  constructor(props) {

    super(props);

    this.state = {text: ''};

  }

  render() {

    return (

      <View style={{padding: 10}}>

        <TextInput

          style={{height: 40}}

          placeholder="Type here to translate!"

          onChangeText={(text) => this.setState({text})}

          value={this.state.text}

        />

         <Button

          title="Press me"

          onPress={() => Alert.alert('Simple Button pressed')}

        />

        <Text style={{padding: 10, fontSize: 42}}>

          {this.state.text.split(' ').map((word) => word && '🍕').join(' ')}

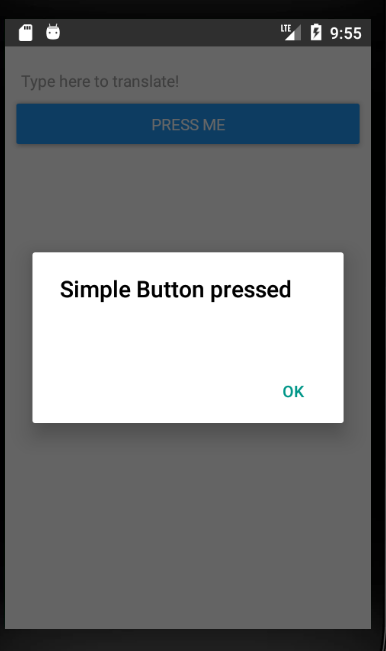
        </Text>

      </View>

    );

  }

}



CheckBox Component:

export default class PRNRadioForm extends Component {

    \_onSelect = ( item ) => {

      console.log(item);

    };

  render() {

    return (

      <View style={styles.container}>

          <View style={{ marginVertical: 10, backgroundColor: "#E7E7E7" }} >

              <CheckboxFormX

                  style={{ width: 350 - 30 }}

                  dataSource={mockData}

                  itemShowKey="label"

                  itemCheckedKey="RNchecked"

                  iconSize={16}

                  formHorizontal={true}

                  labelHorizontal={false}

                  onChecked={(item) => this.\_onSelect(item)}

              />

          </View>

     </View>

    );

  }

}

Radio Button:

import RadioForm, {RadioButton, RadioButtonInput, RadioButtonLabel} from 'react-native';

var radio\_props = [

  {label: 'param1', value: 0 },

  {label: 'param2', value: 1 }

];

var RadioButtonProject = React.createClass({

  getInitialState: function() {

    return {

      value: 0,

    }

  },

  render: function() {

    return (

      <View>

        <RadioForm

          radio\_props={radio\_props}

          initial={0}

          onPress={(value) => {this.setState({value:value})}}

        />

      </View>

    );

  }

});

ProgressBar

VD:

import React, {Component} from 'react';

import {ProgressBarAndroid, StyleSheet, View} from 'react-native';

export default class progress extends Component {

render() {

return (

<View style={styles.container}>

<ProgressBarAndroid />

<ProgressBarAndroid styleAttr="Horizontal" />

<ProgressBarAndroid styleAttr="Horizontal" color="#2196F3" />

<ProgressBarAndroid

styleAttr="Horizontal"

indeterminate={false}

progress={0.5}

/>

</View>

);

}

}

const styles = StyleSheet.create({

container: {

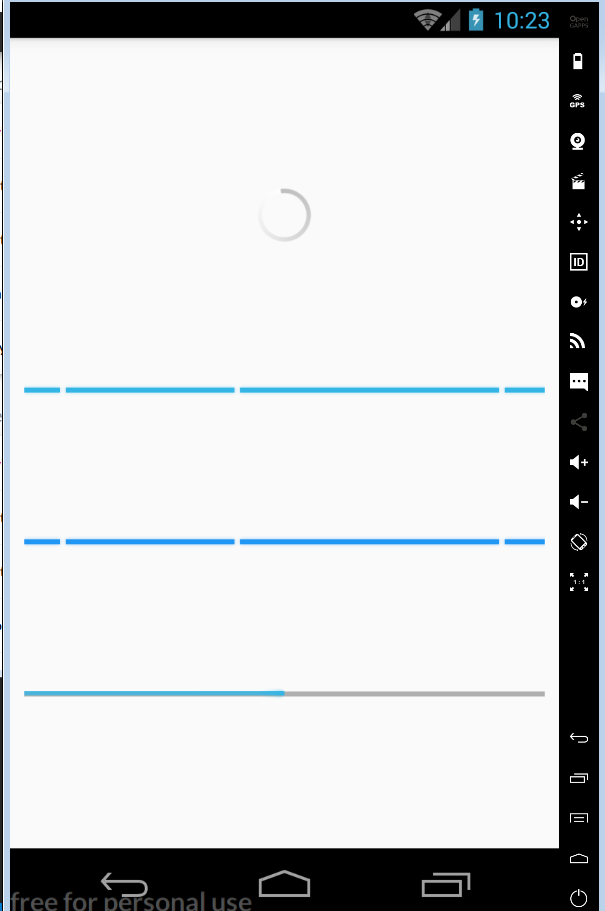
flex: 1,

justifyContent: 'space-evenly',

padding: 10,

},

});



ScrollView

VD:

import React, { Component } from 'react';

import { ScrollView, Image, Text } from 'react-native';

export default class IScrolledDownAndWhatHappenedNextShockedMe extends Component {

render() {

return (

<ScrollView>

<Text style={{fontSize:96}}>Scroll me plz</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}}/>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>If you like</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>Scrolling down</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>What's the best</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>Framework around?</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

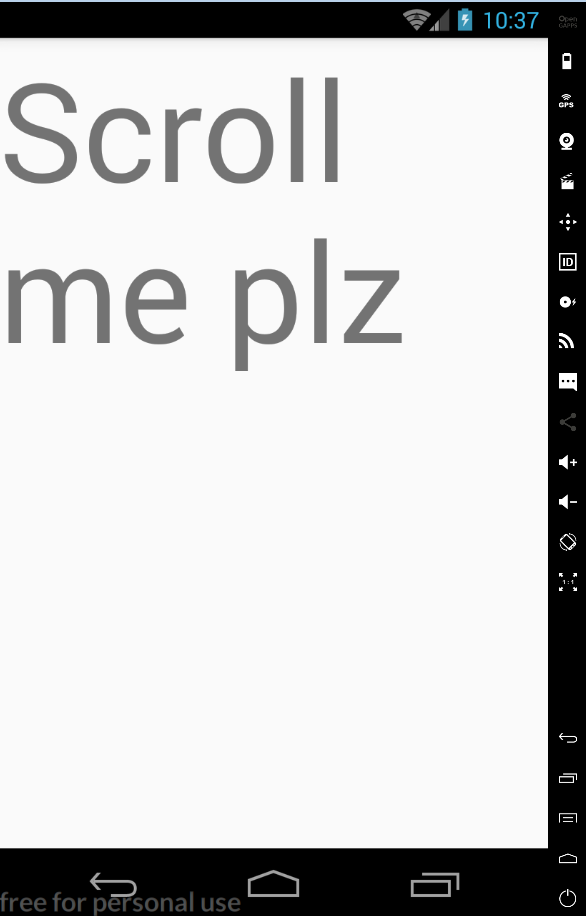
<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:80}}>React Native</Text>

</ScrollView>



ListView

VD:

import React, { Component } from 'react';

import { FlatList, StyleSheet, Text, View } from 'react-native';

export default class FlatListBasics extends Component {

render() {

return (

<View style={styles.container}>

<FlatList

data={[

{key: 'Devin'},

{key: 'Dan'},

{key: 'Dominic'},

{key: 'Jackson'},

{key: 'James'},

{key: 'Joel'},

{key: 'John'},

{key: 'Jillian'},

{key: 'Jimmy'},

{key: 'Julie'},

]}

renderItem={({item}) => <Text style={styles.item}>{item.key}</Text>}

/>

</View>

);

}

}

const styles = StyleSheet.create({

container: {

flex: 1,

paddingTop: 22

},

item: {

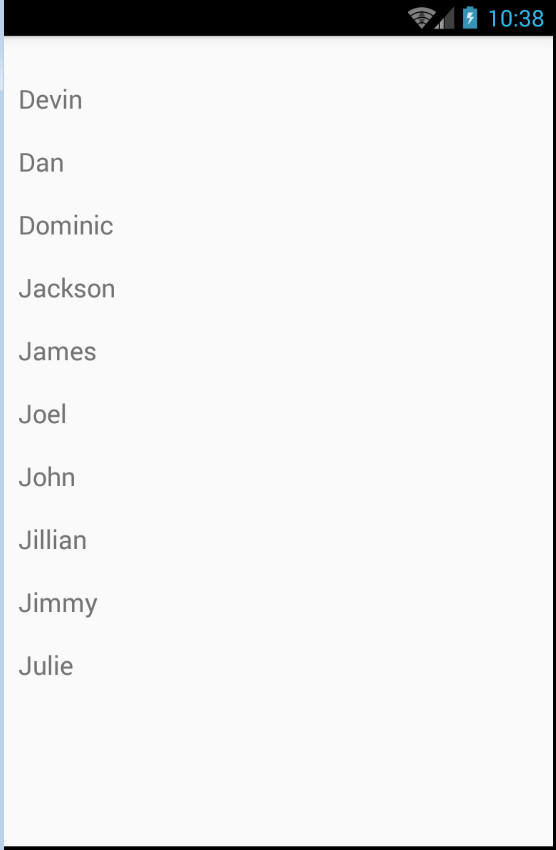
padding: 10,

fontSize: 18,

height: 44,

},

})



Picker

VD:

import React, { Component } from 'react'

import {StyleSheet,View, Text,Picker} from 'react-native'

export default class SwitchExample extends Component {

state = {

choosenIndex: 0

};

render() {

return (

<View style={styles.container}>

<Text style={styles.textStyle}>Swith Example</Text>

<Picker style={styles.pickerStyle}

selectedValue={this.state.language}

onValueChange={(itemValue, itemPosition) =>

this.setState({language: itemValue, choosenIndex: itemPosition})}

>

<Picker.Item label="Java" value="java" />

<Picker.Item label="JavaScript" value="js" />

<Picker.Item label="React Native" value="rn" />

</Picker>

<Text style={styles.textStyle}> {"Index ="+this.state.choosenIndex}</Text>

</View>

);

}

}

const styles = StyleSheet.create ({

container: {

flex: 1,

alignItems: 'center',

justifyContent: 'center',

},

textStyle:{

margin: 24,

fontSize: 25,

fontWeight: 'bold',

textAlign: 'center',

},

pickerStyle:{

height: 150,

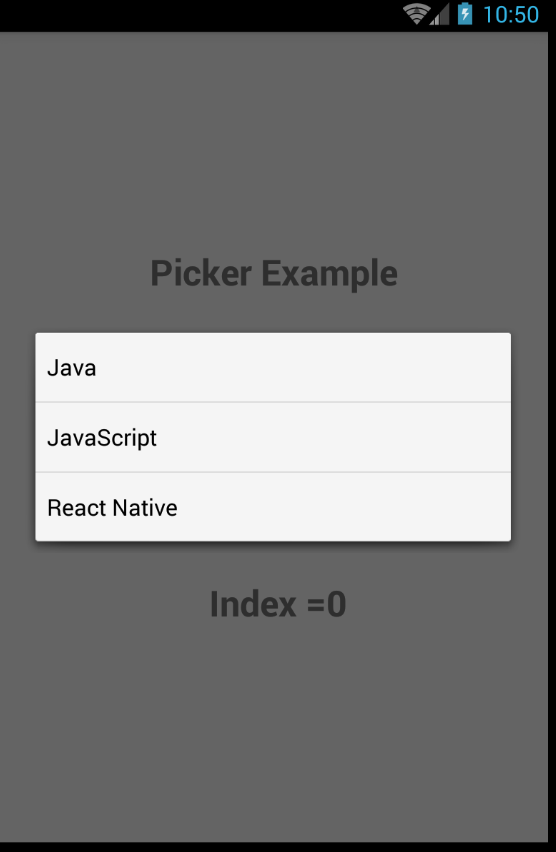
width: "80%",

color: '#344953',

justifyContent: 'center',

}

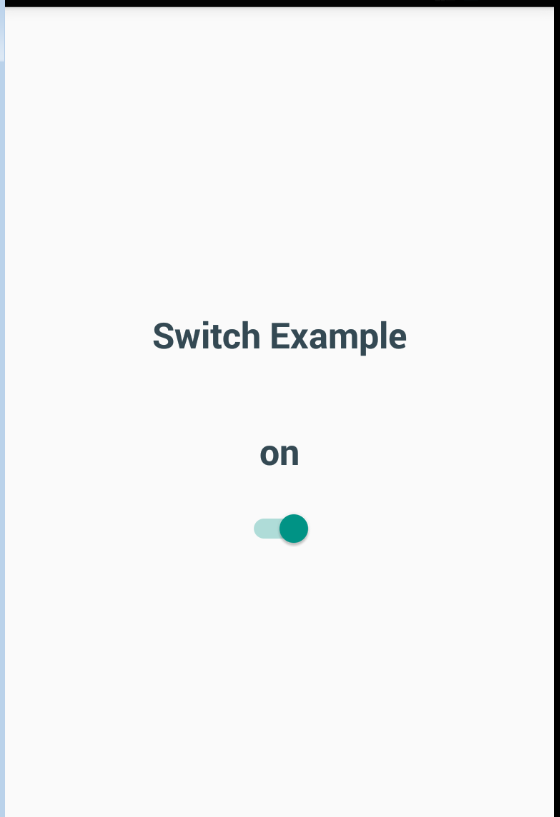
})



Switch

VD:

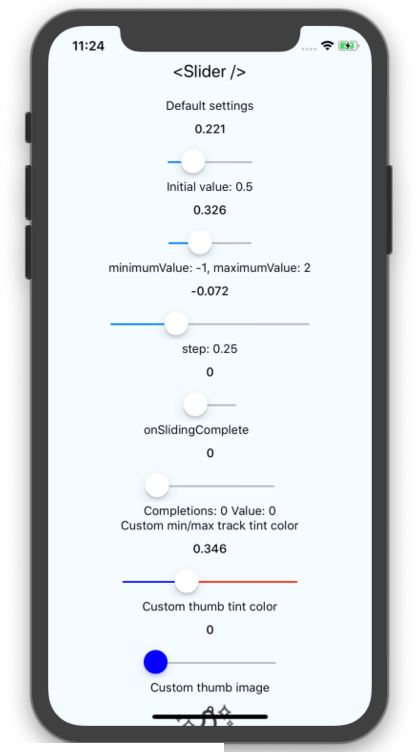
1. **import** React, { Component } from 'react'
2. **import** {StyleSheet, Switch, View, Text} from 'react-native'
4. export **default** **class** SwitchExample **extends** Component {
5. state = {
6. switchValue: **false**
7. };
9. render() {
10. **return** (
11. <View style={styles.container}>
12. <Text style={styles.textStyle}>Switch Example</Text>
13. <Text style={styles.textStyle}>{**this**.state.switchValue ? 'on' :'off'}</Text>
14. <Switch
15. value={**this**.state.switchValue}
16. onValueChange ={(switchValue)=>**this**.setState({switchValue})}/>
17. </View>
18. );
19. }
20. }
21. **const** styles = StyleSheet.create ({
22. container: {
23. flex: 1,
24. alignItems: 'center',
25. justifyContent: 'center',
26. },
27. textStyle:{
28. margin: 24,
29. fontSize: 25,
30. fontWeight: 'bold',
31. textAlign: 'center',
32. color: '#344953'
33. }
34. })



Slider

VD:

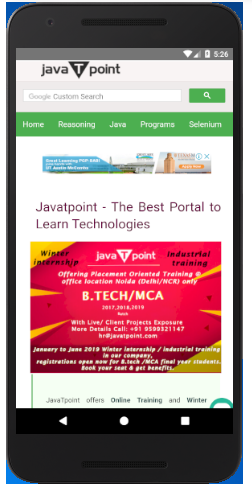
|  |
| --- |
| /\*\* |
|  | \* Copyright (c) Facebook, Inc. and its affiliates. |
|  | \* |
|  | \* This source code is licensed under the MIT license found in the |
|  | \* LICENSE file in the root directory of this source tree. |
|  | \* |
|  | \* @format |
|  | \* @flow |
|  | \*/ |
|  |  |
|  | 'use strict'; |
|  |  |
|  | const React = require('react'); |
|  | const {Text, StyleSheet, View} = require('react-native'); |
|  | const Slider = require('@react-native-community/slider'); |
|  |  |
|  | class SliderExample extends React.Component<$FlowFixMeProps, $FlowFixMeState> { |
|  | static defaultProps = { |
|  | value: 0, |
|  | }; |
|  |  |
|  | state = { |
|  | value: this.props.value, |
|  | }; |
|  |  |
|  | render() { |
|  | return ( |
|  | <View> |
|  | <Text style={styles.text}> |
|  | {this.state.value && +this.state.value.toFixed(3)} |
|  | </Text> |
|  | <Slider |
|  | {...this.props} |
|  | onValueChange={value => this.setState({value: value})} |
|  | /> |
|  | </View> |
|  | ); |
|  | } |
|  | } |
|  |  |
|  | class SlidingStartExample extends React.Component< |
|  | $FlowFixMeProps, |
|  | $FlowFixMeState, |
|  | > { |
|  | state = { |
|  | slideStartingValue: 0, |
|  | slideStartingCount: 0, |
|  | }; |
|  |  |
|  | render() { |
|  | return ( |
|  | <View> |
|  | <SliderExample |
|  | {...this.props} |
|  | onSlidingStart={value => |
|  | this.setState({ |
|  | slideStartingValue: value, |
|  | slideStartingCount: this.state.slideStartingCount + 1, |
|  | }) |
|  | } |
|  | /> |
|  | <Text> |
|  | Starts: {this.state.slideStartingCount} Value:{' '} |
|  | {this.state.slideStartingValue} |
|  | </Text> |
|  | </View> |
|  | ); |
|  | } |
|  | } |
|  |  |
|  | class SlidingCompleteExample extends React.Component< |
|  | $FlowFixMeProps, |
|  | $FlowFixMeState, |
|  | > { |
|  | state = { |
|  | slideCompletionValue: 0, |
|  | slideCompletionCount: 0, |
|  | }; |
|  |  |
|  | render() { |
|  | return ( |
|  | <View> |
|  | <SliderExample |
|  | {...this.props} |
|  | onSlidingComplete={value => |
|  | this.setState({ |
|  | slideCompletionValue: value, |
|  | slideCompletionCount: this.state.slideCompletionCount + 1, |
|  | }) |
|  | } |
|  | /> |
|  | <Text> |
|  | Completions: {this.state.slideCompletionCount} Value:{' '} |
|  | {this.state.slideCompletionValue} |
|  | </Text> |
|  | </View> |
|  | ); |
|  | } |
|  | } |
|  |  |
|  | const styles = StyleSheet.create({ |
|  | text: { |
|  | fontSize: 14, |
|  | textAlign: 'center', |
|  | fontWeight: '500', |
|  | margin: 10, |
|  | }, |
|  | }); |
|  |  |
|  | exports.title = '<Slider>'; |
|  | exports.displayName = 'SliderExample'; |
|  | exports.description = 'Slider input for numeric values'; |
|  | exports.examples = [ |
|  | { |
|  | title: 'Default settings', |
|  | render(): React.Element<any> { |
|  | return <SliderExample />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Initial value: 0.5', |
|  | render(): React.Element<any> { |
|  | return <SliderExample value={0.5} />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'minimumValue: -1, maximumValue: 2', |
|  | render(): React.Element<any> { |
|  | return <SliderExample minimumValue={-1} maximumValue={2} />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'step: 0.25', |
|  | render(): React.Element<any> { |
|  | return <SliderExample step={0.25} />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'onSlidingStart', |
|  | render(): React.Element<any> { |
|  | return <SlidingStartExample />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'onSlidingComplete', |
|  | render(): React.Element<any> { |
|  | return <SlidingCompleteExample />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Custom min/max track tint color', |
|  | render(): React.Element<any> { |
|  | return ( |
|  | <SliderExample |
|  | minimumTrackTintColor={'blue'} |
|  | maximumTrackTintColor={'red'} |
|  | value={0.5} |
|  | /> |
|  | ); |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Custom thumb tint color', |
|  | render(): React.Element<any> { |
|  | return <SliderExample thumbTintColor={'blue'} />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Custom thumb image', |
|  | platform: 'ios', |
|  | render(): React.Element<any> { |
|  | return <SliderExample thumbImage={require('./uie\_thumb\_big.png')} />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Custom track image', |
|  | platform: 'ios', |
|  | render(): React.Element<any> { |
|  | return <SliderExample trackImage={require('./slider.png')} />; |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Custom min/max track image', |
|  | platform: 'ios', |
|  | render(): React.Element<any> { |
|  | return ( |
|  | <SliderExample |
|  | minimumTrackImage={require('./slider-left.png')} |
|  | maximumTrackImage={require('./slider-right.png')} |
|  | /> |
|  | ); |
|  | }, |
|  | }, |
|  | { |
|  | title: 'Inverted slider direction', |
|  | platform: 'ios', |
|  | render(): React.Element<any> { |
|  | return <SliderExample value={0.6} inverted />; |
|  | }, |
|  | }, |
|  | ]; |



WebView

VD:

1. **import** React, { Component } from 'react'
2. **import** {
3. View,WebView,StyleSheet,AppRegistry
4. } from 'react-native'
6. export **default** **class** ActivityIndicatorDemo **extends** Component {
7. render() {
8. **return** (
9. <View style = {styles.container}>
10. <WebView
11. source = {{ uri:'https://www.javatpoint.com' }}
12. />
13. </View>
14. )
15. }
16. }
17. **const** styles = StyleSheet.create({
18. container: {
19. flex: 1,
20. }
21. })
23. AppRegistry.registerComponent('App', () => ActivityIndicatorDemo)



ViewPager

VD:

import ViewPager from "@react-native-community/viewpager";

class MyPager extends React.Component {

render() {

return (

<ViewPager

style={styles.viewPager}

initialPage={0}>

<View key="1">

<Text>First page</Text>

</View>

<View key="2">

<Text>Second page</Text>

</View>

</ViewPager>

);

}

}

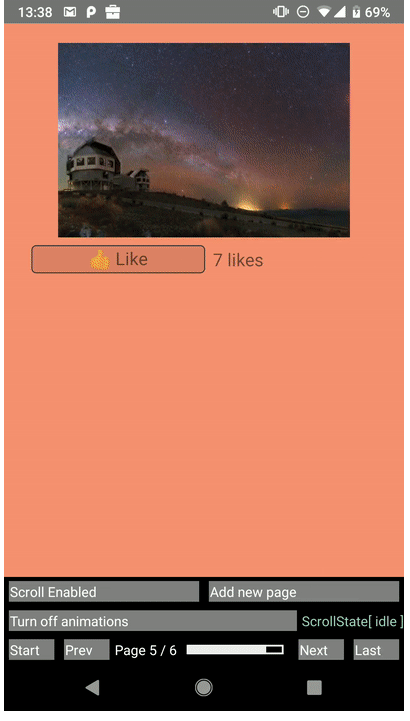
const styles = StyleSheet.create({

viewPager: {

flex: 1

},

})



DrawerLayout

VD