1. Implement WebView, show Activity indicator when WebView is loading

import React, {Component} from 'react';

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

import {WebView} from 'react-native-webview';

class webviewc extends Component {

  constructor(props) {

    super(props);

    this.state = {

      isLoading: true

    }

  }

  render() {

    return(

      <SafeAreaView style={styles.container}>

        <WebView

          onLoadStart={() => {

            this.setState({isLoading: true});

          }}

          onLoadEnd = {() => {

            this.setState({isLoading: false});

          }}

          source={{uri: 'https://www.youtube.com' }}

        />

        {this.state.isLoading ? (

          <View style={styles.activity}>

            <ActivityIndicator

              size={'large'}

              color={'green'}

            />

          </View>

        ): null}

      </SafeAreaView>

    )

  }

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

  },

  activity: {

    justifyContent: 'center',

    alignItems: 'center',

    width: '100%',

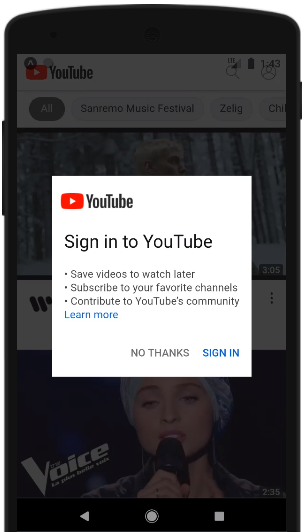
    height: '100%'

  }

});

export default webviewc;

OUTPUT:



2.) Use a switch to show hide the Slider and below slider show the slider value

import React, { useState } from "react";

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

import Slider from '@react-native-community/slider';

const slideandswitch = () => {

  const [isEnabled, setIsEnabled] = useState(false);

  const [sliderValue, setValue] = useState(1);

  const toggleSwitch = () => {

    setIsEnabled(previousState => !previousState)

    setValue(0);

  }

  return (

    <View style={styles.container}>

      <Switch

        style={styles.Switch}

        trackColor = {{ false: "#36402f", true: "#3498db" }}

        thumbColor = {isEnabled ? "purple" : "white"}

        onValueChange={toggleSwitch}

        value={isEnabled}

      />

      {

        isEnabled ? (

            <View style={styles.xyz}>

              <Slider

                minimumValue = {0}

                maximumValue = {100}

                value={0}

                step={1}

                style={styles.SliderMain}

                onSlidingComplete={setValue}

            />

              <Text style={styles.Slid}>Slider Value: {sliderValue}</Text>

            </View>

        ) : null

      }

    </View>

  )

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    alignItems: 'center',

    justifyContent: 'center',

    backgroundColor: '#daf6f7'

  },

  Switch: {

    position: 'absolute',

    top: 300,

    marginTop:40,

    transform:[{scale:1.0}]

  },

  xyz: {

    marginVertical: 50,

  },

  SliderMain : {

    width: 300,

    height: 40,

  },

  Slid : {

    color: 'black',

    textAlign: 'center',

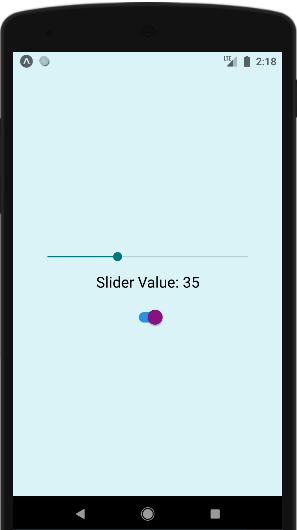
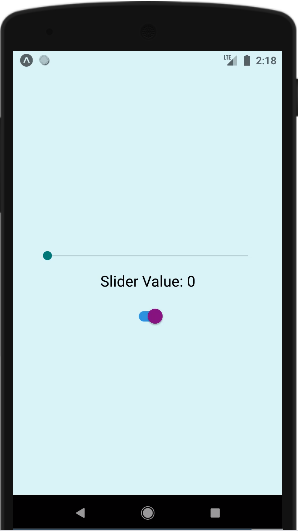
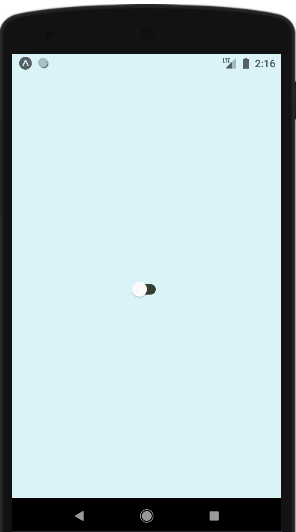
    fontSize: 20,

    marginBottom:50

  }

})

export default slideandswitch;



3). Using api https://reqres.in/api/users?page=1 Render a Flatlist and implement pagination and pull to refresh show activity indicator below the flatlist when api is being called for the next page, use RefreshControl to show Activity indicator on top in case of pull to refresh

import React from "react";

import {

StyleSheet,

View,

ActivityIndicator,

FlatList,

Text,

TouchableOpacity,

ButtonProps,

Button,

Image,

} from "react-native";

export default class App extends React.Component {

constructor(props) {

super(props);

this.state = {

loading: true,

dataSource:[] };

}

componentDidMount(){

fetch("https://reqres.in/api/users")

.then(response => response.json())

.then((responseData)=> {

this.setState({

loading: false,

dataSource: responseData.data

})

})

}

FlatListItemSeparator = () => {

return (

<View style={{

height: .5,

width:"100%",

backgroundColor:"rgba(0,0,0,0.5)",

}}

/>

);

}

      renderItem=(data)=>

      <TouchableOpacity style={styles.list}>

       <Text style={styles.lightText}>id : {data.item.id}</Text>

       <Text style={styles.lightText}>email : {data.item.email}</Text>

       <Image style={styles.avatar} source={{uri:`${data.item.avatar}`}}/>

       <Text style={styles.lightText}> {data.item.first\_name}</Text>

       <Text style={styles.lightText}> {data.item.last\_name}</Text>

       </TouchableOpacity>

render(){

if(this.state.loading){

return(

<View style={styles.loader}>

<ActivityIndicator size="large" color="black"/>

</View>

)}

return(

<View style={styles.container}>

<View>

<Text> Employee Details</Text>

</View>

<FlatList

data= {this.state.dataSource}

renderItem= {item=> this.renderItem(item)}

keyExtractor= {item=>item.id.toString()}

/>

</View>

)}

}

const styles = StyleSheet.create({

container: {

flex: 1,

backgroundColor: "white"

},

avatar:{

  height:50,

  width:50,

  alignContent:"center",

  Text:"center"

},

loader:{

flex: 1,

justifyContent: "center",

alignItems: "center",

backgroundColor: "black"

},

list:{

paddingVertical: 2,

margin: 5,

backgroundColor: "#FFBF00"

}

});

