**Basic Components:**

**View:**

* View is the most common element in React Native. You can consider it as an equivalent of the **div** element used in web development.
* The most fundamental component for building a UI, View is a container that supports layout with flexbox, Style, Some Touch Handling.
* When you want to nest more elements inside the parent element, both parent and child can be **View**. It can have as many children as you want.
* View is designed to be nested inside other views and can have 0 to many children of any type.
* Views are designed to be used with stylesheet for clarity and performance, although inline styles are also supported.

import React from 'react'

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

export const CreateAcc = () => {

    return (

       <View>

           <Text>Hi Venkatesh !!!</Text>

       </View>

    );

};

**Text:**

* A React component for displaying text
* Text supports nesting, styling, and touch handling.
* Text displays words and characters at various sizes.

import React, {useState} from 'react'

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

export const CreateAcc = () => {

    return (

        <View style = {styles.container}>

         <Text style = {styles.text}>

            <Text style = {styles.capitalLetter}>

               T

            </Text>

            <Text>

            his  is  <Text style = {styles.wordBold}>Venkatesh Babu </Text> from

            </Text>

            {"\b"}

            <Text style = {styles.italicText}>

                Srivilliputtur

            </Text>

         </Text>

      </View>

   )

}

const styles = StyleSheet.create ({

   container: {

      alignItems: 'center',

      marginTop: 100,

      padding: 20

   },

   text: {

      color: '#41cdf4',

   },

   capitalLetter: {

      color: 'red',

      fontSize: 20

   },

   wordBold: {

      fontWeight: 'bold',

      color: 'black'

   },

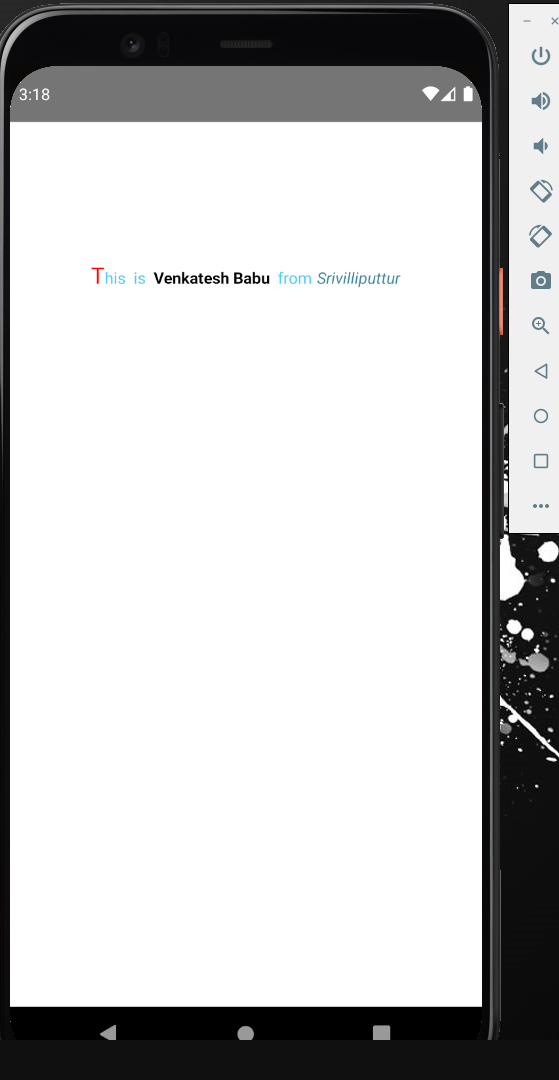
   italicText: {

      color: '#37859b',

      fontStyle: 'italic'

   },

})

****

**TextInput:**

* TextInput is used to collect data from users.
* For inputting text into the app via a keyboard.
* Props provide configurability for several features, such as auto-correction, auto-capitalization, placeholder text e.t.c
* For creating a TextInput in react native we have to import the TextInput component from React Native.

import {TextInput} from 'react-native'

**Ex:**

import React from 'react';

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

export const Login = () => {

    return (

        <View>

            <Text style= {{color: 'black’, marginLeft:50, marginTop:60, fontSize:15}}>Email</Text>

            <TextInput style= {{color:'rgba (167,167,167,1)',marginLeft:50, marginTop:5,fontSize:14,margin:10,borderRadius:10,borderWidth:1}} placeholder="Enter your EmailId" ></TextInput>

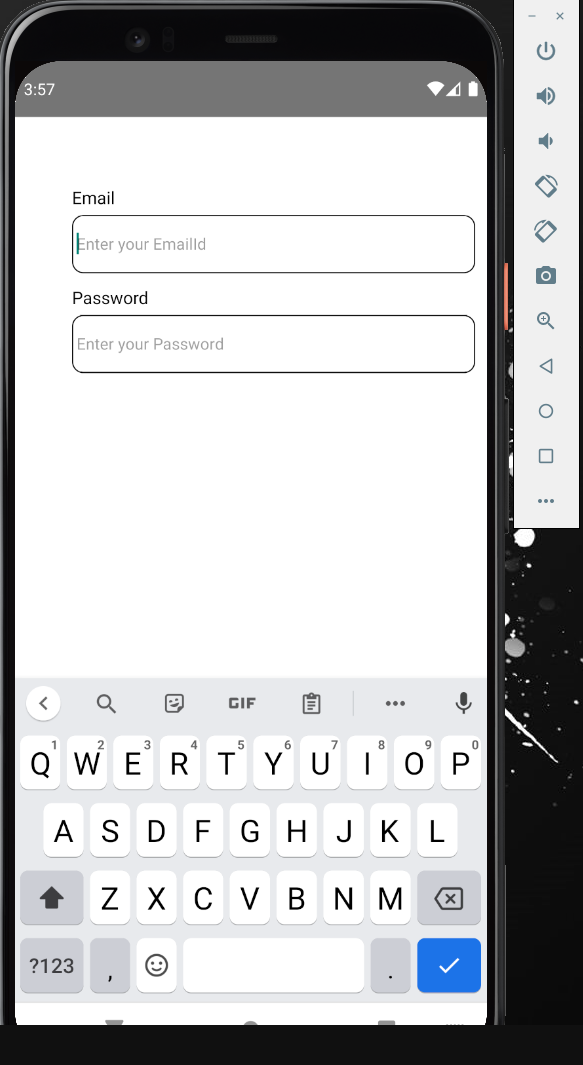
            <Text style={{color: 'black',marginLeft:50,marginTop:1,fontSize:15}}>Password</Text>

            <TextInput style={{color: 'rgba(167,167,167,1)',marginLeft:50,marginTop:5,fontSize:14,margin:10,borderRadius:10,borderWidth:1}} placeholder="Enter your Password" ></TextInput>

        </View>

    );

};

****

**Image:**

* A React component for displaying different types of images, including network images, static resources, temporary local images, and images from local disk.
* Note that for network and data images, you will need to manually specify the dimensions of your image!

**Adding Image:**

* Let us create a new folder **image** inside the **src** folder. We will add our image (**myImage.png**) inside this folder.
* Displaying images in React Native starts with importing **Image** API from **‘react-native’.**

**@Local Image:**

* First to display the image we have to create an **‘Image’s** folder at the root of your project folder, add our Images (**myImage.png**) inside this folder.
* You can easily add the custom styling in Image component. As you can see, we have set the **width, height** and **margin-bottom** properties to style Image component.

**Ex:**

import React from 'react';

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

export const Login = () => {

    return (

        <View>

          <Image

            style={{ width: 100, height: 100, marginBottom: 15 }}

            source={require('./Images/flower.png')}

          />

        </View>

      );

    }

**@Displaying Network Images:**

* To display images via Network requests, we have to use source property instead of require tag.
* The same way you can add the custom CSS properties to style the Image Component in React Native.
* It is recommended to define the **width** and the **height** for network images.

import React from 'react';

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

export const Login = () => {

    return (

        <View>

          <Image

            style={{ width: 100, height: 100}}

            source={{uri: ('https://cdn.pixabay.com/photo/2014/11/29/19/51/identity-550820\_1280.jpg')}}

          />

        </View>

      );

    }

**@ScrollView:**

* The ScrollView is a generic scrollable container, which scrolls multiple child components and views inside it.
* In the ScrollView, we can scroll the components in **both directions vertically and horizontally**.
* Ever have the situation where you sometimes had content that was shorter than the screen size and didn’t require scrolling but occasionally had content taller than the screen size, thus necessitating scroll to allow the user to see all the content.

import React from 'react';

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

export const Login = () => {

    return (

        <ScrollView>

        <Text style={styles.Venkatesh}>Hi Venkatesh</Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        <Text style={styles.Venkatesh}>Hi Venkatesh </Text>

        </ScrollView>

      );

    }

    const styles = StyleSheet.create({

        Venkatesh: {

          margin: 20,

          backgroundColor: 'orange',

          margin: 10,

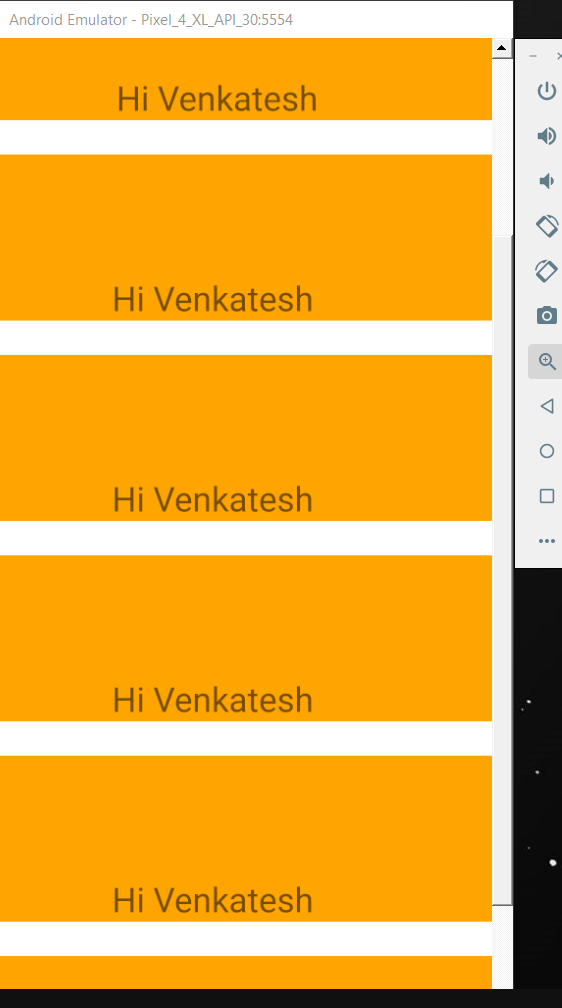
          textAlign: 'center',

          fontSize: 20,

          paddingTop: 70,

        }

      })

****

**@StyleSheet:**

* A StyleSheet is an abstraction similar to CSS Stylesheet’s.
* There are two types of StyleSheet:
* **Inline stylesheet’s:** Inline styles are implemented specific to each component through the style tag. Style property can be utilized to add inline style to components. But this approach is not considered as best practice as it makes code complex.
* **External Stylesheets:** These style sheets are implemented using separate JavaScript file which can be imported into our existing code. External Stylesheets are standalone files and advantage associated with these files is that they can be reused multiple times.
* **The following things to be kept in mind while using react native stylesheet’s:**
* Avoid creation of new stylesheet’s always try to reuse existing stylesheets.
* Always keep style sheet code outside render method to provide better performance.
* Stylesheet is rendered only once inside render method.

**@Code quality tips:**

* By moving styles away from the render function, you're making the code easier to understand.
* Naming the styles is a good way to add meaning to the low level components in the render function.
* We will import the **StyleSheet**. At the bottom of the file, we will create our stylesheet and assign it to the **styles** constant.
* **Note that our styles are in camelCase and we do not use px or % for styling.**
* To apply styles to our text, we need to add **style = {styles.myText}** property to the **Text** element.

import React from 'react';

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

export const Login = () => {

    return (

        <View>

        <Text style={styles.Venkatesh}>Hi Venkatesh</Text>

        </View>

      );

    }

    const styles = StyleSheet.create ({

        Venkatesh: {

           marginTop: 20,

           textAlign: 'center',

           color: 'blue',

           fontWeight: 'bold',

           fontSize: 20

        }

     })

