**Assignment 4**

**Name: Ayesha Siddiqua**

**Registration Number: FA20-BCS-072**

**Subject: Mobile Application Development**

**Submitted To: Mr. Muhammad Kamran**

**Question:** Extend the Assignment 3, and store the information submitted by user in signup form inside local storage.And then in Login form retrieve that information and if valid information provided by the user, redirect him to profile screen.

**Code:**

import React, { useState } from 'react';

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

import { NavigationContainer } from '@react-navigation/native';

import { createStackNavigator } from '@react-navigation/stack';

const Stack = createStackNavigator();

export default function App() {

  return (

    <NavigationContainer>

      <Stack.Navigator initialRouteName="SignUp">

        <Stack.Screen name="SignUp" component={SignUpPage} />

        <Stack.Screen name="SignIn" component={SignInPage} />

        <Stack.Screen name="Profile" component={ProfilePage} />

      </Stack.Navigator>

    </NavigationContainer>

  );

}

function SignUpPage({ navigation }) {

  const [firstName, setFirstName] = useState('');

  const [lastName, setLastName] = useState('');

  const [email, setEmail] = useState('');

  const [repeatEmail, setRepeatEmail] = useState('');

  const [password, setPassword] = useState('');

  const [repeatPassword, setRepeatPassword] = useState('');

  const [gender, setGender] = useState('');

  const [age, setAge] = useState('');

  const [country, setCountry] = useState('');

  const [city, setCity] = useState('');

  const handleSignUp = async () => {

    // Check if all fields are filled

    if (

      firstName !== '' &&

      lastName !== '' &&

      email !== '' &&

      repeatEmail !== '' &&

      password !== '' &&

      repeatPassword !== '' &&

      gender !== '' &&

      age !== '' &&

      country !== '' &&

      city !== ''

    ) {

      // Check if email and repeat email match

      if (email === repeatEmail) {

        // Check if password and repeat password match

        if (password === repeatPassword) {

          // Create user object with the form data

          const user = {

            firstName,

            lastName,

            email,

            password,

            gender,

            age,

            country,

            city,

          };

          try {

            // Store the user object in local storage

            await AsyncStorage.setItem('user', JSON.stringify(user));

            navigation.navigate('Profile', { user }); // Navigate to Profile screen and pass user object

          } catch (error) {

            console.log('Error storing user data:', error);

          }

        } else {

          alert('Password and Repeat Password do not match');

        }

      } else {

        alert('Email and Repeat Email do not match');

      }

    } else {

      alert('Please fill in all fields');

    }

  };

  return (

    <View style={styles.container}>

      <Text style={{ color: 'blue', fontSize: 40, fontWeight: 'bold', marginBottom: 10, marginRight: 210 }}>

        Sign Up

      </Text>

      <View style={styles.inputRow}>

        <TextInput

          style={[styles.textInput, { marginRight: 30, marginTop: 30 }]}

          placeholder="First Name"

          value={firstName}

          onChangeText={setFirstName}

        />

        <TextInput

          style={[styles.textInput, { marginLeft: 0, marginTop: 30 }]}

          placeholder="Last Name"

          value={lastName}

          onChangeText={setLastName}

        />

      </View>

      <View style={styles.inputRow}>

        <TextInput

          style={[styles.textInput, { marginRight: 30 }]}

          placeholder="Email"

          value={email}

          onChangeText={setEmail}

        />

        <TextInput

          style={[styles.textInput, { marginRight: 0 }]}

          placeholder="Repeat Email"

          value={repeatEmail}

          onChangeText={setRepeatEmail}

        />

      </View>

      <View style={styles.inputRow}>

        <TextInput

          style={[styles.textInput, { marginRight: 30 }]}

          placeholder="Password"

          secureTextEntry

          value={password}

          onChangeText={setPassword}

        />

        <TextInput

          style={[styles.textInput, { marginRight: 0 }]}

          placeholder="Repeat Password"

          secureTextEntry

          value={repeatPassword}

          onChangeText={setRepeatPassword}

        />

      </View>

      <View style={styles.inputRow}>

        <TextInput

          style={[styles.textInput, { marginRight: 30 }]}

          placeholder="Gender"

          value={gender}

          onChangeText={setGender}

        />

        <TextInput

          style={[styles.textInput, { marginRight: 0 }]}

          placeholder="Age"

          value={age}

          onChangeText={setAge}

        />

      </View>

      <View style={styles.inputRow}>

        <TextInput

          style={[styles.textInput, { marginRight: 30 }]}

          placeholder="Country"

          value={country}

          onChangeText={setCountry}

        />

        <TextInput

          style={[styles.textInput, { marginRight: 0 }]}

          placeholder="City"

          value={city}

          onChangeText={setCity}

        />

      </View>

      <TouchableOpacity style={styles.button} onPress={handleSignUp}>

        <Text style={styles.buttonText}>Sign Up</Text>

      </TouchableOpacity>

      <TouchableOpacity onPress={() => navigation.navigate('SignIn')}>

        <Text style={{ color: 'blue', fontSize: 16, marginTop: 20 }}>Already have an account? Sign In</Text>

      </TouchableOpacity>

    </View>

  );

}

function SignInPage({ navigation }) {

  const [email, setEmail] = useState('');

  const [password, setPassword] = useState('');

  const handleSignIn = async () => {

    try {

      // Retrieve user data from local storage

      const storedUser = await AsyncStorage.getItem('user');

      if (storedUser) {

        const user = JSON.parse(storedUser);

        // Check if provided email and password match the stored user data

        if (email === user.email && password === user.password) {

          navigation.navigate('Profile', { user }); // Navigate to Profile screen and pass user object

        } else {

          alert('Invalid email or password');

        }

      } else {

        alert('No user found. Please sign up first.');

      }

    } catch (error) {

      console.log('Error retrieving user data:', error);

    }

  };

  return (

    <View style={styles.container}>

      <Text style={{ color: 'blue', fontSize: 40, fontWeight: 'bold', marginBottom: 10, marginRight: 210 }}>Sign In</Text>

      <TextInput

        style={styles.textInput}

        placeholder="Email"

        value={email}

        onChangeText={setEmail}

      />

      <TextInput

        style={styles.textInput}

        placeholder="Password"

        secureTextEntry

        value={password}

        onChangeText={setPassword}

      />

      <TouchableOpacity style={styles.button} onPress={handleSignIn}>

        <Text style={styles.buttonText}>Sign In</Text>

      </TouchableOpacity>

      <TouchableOpacity onPress={() => navigation.navigate('SignUp')}>

        <Text style={{ color: 'blue', fontSize: 16, marginTop: 20 }}>Don't have an account? Sign Up</Text>

      </TouchableOpacity>

    </View>

  );

}

function ProfilePage({ route }) {

  // Retrieve user information from the route params

  const { user } = route.params;

  return (

    <View style={styles.container}>

      <Text style={{ color: 'blue', fontSize: 40, fontWeight: 'bold', marginBottom: 10 }}>Profile</Text>

      <Text style={styles.profileText}>First Name: {user.firstName}</Text>

      <Text style={styles.profileText}>Last Name: {user.lastName}</Text>

      <Text style={styles.profileText}>Email: {user.email}</Text>

      <Text style={styles.profileText}>Gender: {user.gender}</Text>

      <Text style={styles.profileText}>Age: {user.age}</Text>

      <Text style={styles.profileText}>Country: {user.country}</Text>

      <Text style={styles.profileText}>City: {user.city}</Text>

    </View>

  );

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    alignItems: 'center',

    justifyContent: 'center',

    paddingHorizontal: 30,

  },

  inputRow: {

    flexDirection: 'row',

    justifyContent: 'space-between',

    width: '100%',

  },

  textInput: {

    borderWidth: 1,

    borderColor: 'black',

    borderRadius: 5,

    padding: 10,

    width: '48%',

  },

  button: {

    backgroundColor: 'blue',

    borderRadius: 5,

    padding: 10,

    marginTop: 20,

  },

  buttonText: {

    color: 'white',

    fontSize: 16,

    fontWeight: 'bold',

    textAlign: 'center',

  },

  profileText: {

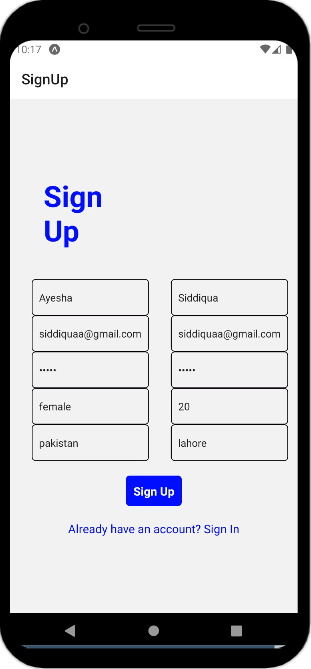
    fontSize: 16,

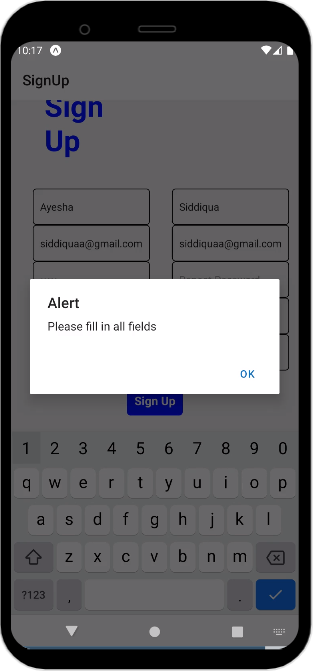
    marginBottom: 5,

  },

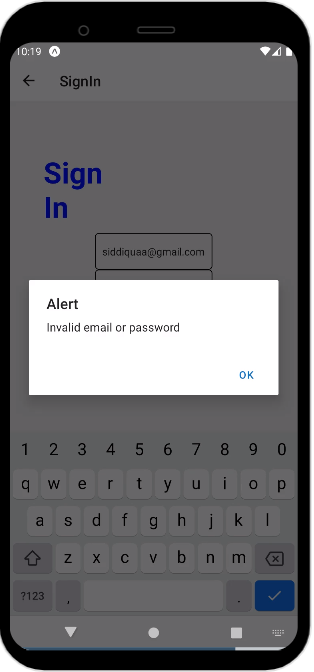
});

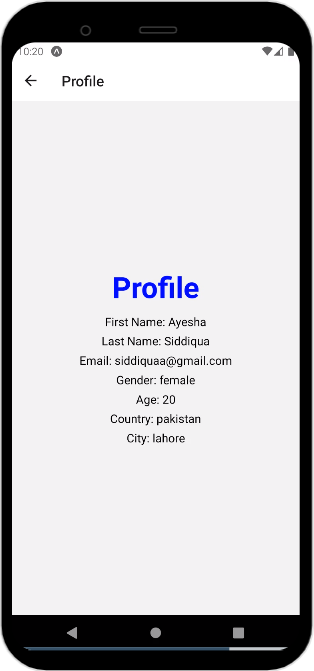
**Output:**

****

****

****

****

****