**Solution: Configure Tab Navigation**

**Overview**

In this reading, you will review the solution code for the exercise.

The code block below outlines the outcome of the exercise in its entirety. All the other code is available in the starter code for this exercise.

**App.js**

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

import \* as React from 'react';

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

import LittleLemonHeader from './components/LittleLemonHeader';

import LittleLemonFooter from './components/LittleLemonFooter';

import WelcomeScreen from './WelcomeScreen';

import LoginScreen from './LoginScreen';

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

import Ionicons from '@expo/vector-icons/Ionicons';

import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';

const Tab = createBottomTabNavigator();

export default function App() {

  return (

    <>

      <NavigationContainer>

        <View style={styles.container}>

          <LittleLemonHeader />

          <Tab.Navigator

            screenOptions={({ route }) => ({

              tabBarIcon: ({size}) => {

                let iconName;

                if (route.name === 'Welcome') {

                  iconName = 'ios-home';

                } else if (route.name === 'Login') {

                  iconName = 'ios-enter';

                }

                return <Ionicons name={iconName} size={size} />;

              },

            })}

            initialRouteName="Login">

            <Tab.Screen name="Welcome" component={WelcomeScreen} />

            <Tab.Screen name="Login" component={LoginScreen} />

          </Tab.Navigator>

        </View>

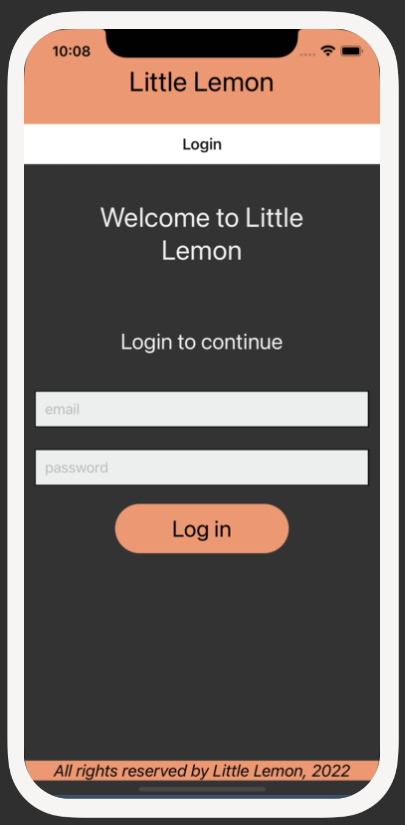
        <View style={styles.footerContainer}>

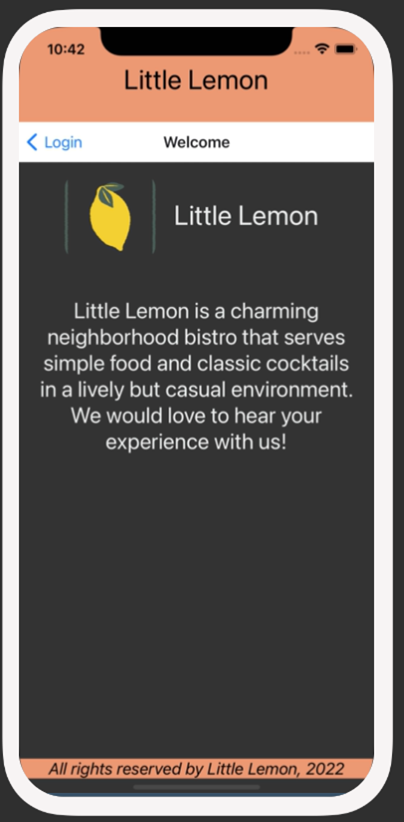
          <LittleLemonFooter />

        </View>

      </NavigationContainer>

The code above would look like the following on the emulator:





**Procedure**

Next, review the breakdown of the individual steps below.

**Step 1: Install tab navigation**

Your first step is to ensure that the dependencies are installed correctly. To check this, open the **package.json** file and verify that it displays the dependencies shown below. If any are missing, make sure to install the React Navigation package as detailed in [**Installation and Setup of React Navigation**](https://www.coursera.org/learn/react-native-course/item/cRZeE), the Stack Navigator library as detailed in Creating and [**Configuring Native Stack Navigator**](https://www.coursera.org/learn/react-native-course/supplement/DEL0E/creating-and-configuring-native-stack-navigator), and the **bottom-tabs** package as explained in [**Exploring Tab Navigation**](https://www.coursera.org/learn/react-native-course/supplement/7TnkG/exploring-tab-navigation).

1

2

3

4

5

6

7

8

9

10

11

12

13

14

{

  "dependencies": {

    "expo-constants": "~13.2.4",

    "@expo/vector-icons": "^13.0.0",

    "react-native-paper": "4.9.2",

    "react-native-screens": "~3.15.0",

    "@react-navigation/native": "6.0.13",

    "@react-navigation/native-stack": "6.9.0",

    "react-native-safe-area-context": "4.3.1",

    "@react-navigation/bottom-tabs": "6.0.0",

    "@expo/vector-icons/Ionicons": "^13.0.0"

  }

}

**Step 2: Instantiate createBottomTabNavigator**

In this step, you will include the necessary imports and instantiate the tab navigator in your **App.js** file as follows.

1

2

3

import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';

const Tab = createBottomTabNavigator();

**Step 3: Setup Tab Navigator and Tab Screens with routes**

With all the initial work done, your final step is to set up the routes within the tab as follows. You can also provide a default route to start with, in this case, it is the **Login** screen. And you are all done setting up routes!

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

const Tab = createBottomTabNavigator();

export default function App() {

  return (

    <>

      <NavigationContainer>

        <View style={styles.container}>

          <LittleLemonHeader />

          <Tab.Navigator

            initialRouteName="Login">

            <Tab.Screen name="Welcome" component={WelcomeScreen} />

            <Tab.Screen name="Login" component={LoginScreen} />

          </Tab.Navigator>

        </View>

        <View style={styles.footerContainer}>

          <LittleLemonFooter />

        </View>

      </NavigationContainer>

    </>

  );

}

**Step 4: Configure icons for each Tab**

In this step, you will import **Ionicons** and use it to set up the tab bar icons as follows. We have used the **ios-home** and **ios-enter** icons for the tabs.

**Import:**

1

import Ionicons from '@expo/vector-icons/Ionicons';

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

export default function App() {

  return (

    <>

      <NavigationContainer>

        <View style={styles.container}>

          <LittleLemonHeader />

          <Tab.Navigator

            screenOptions={({ route }) => ({

              tabBarIcon: ({size}) => {

                let iconName;

                if (route.name === 'Welcome') {

                  iconName = 'ios-home';

                } else if (route.name === 'Login') {

                  iconName = 'ios-enter';

                }

                return <Ionicons name={iconName} size={size} />;

              },

            })}

            initialRouteName="Login">

            <Tab.Screen name="Welcome" component={WelcomeScreen} />

            <Tab.Screen name="Login" component={LoginScreen} />

          </Tab.Navigator>

        </View>

        <View style={styles.footerContainer}>

          <LittleLemonFooter />

        </View>

      </NavigationContainer>

    </>

  );

}