

Using Navigation among the Screens in React Native using Class Components

When we want to visit from one screen to another screen in the app, it is called as screen based navigation.

To achieve the screen based navigation in React Native we need to install two packages in our project `@react-navigation/native` and `@react-navigation/stack`.

To navigate we need to call `navigation.navigate()` function

To try the sample example create a new project using `create-react-native-app`

```
create-react-native-app NavigationDemo
```

Now inside the folder

```
cd NavigationDemo
```

Now install the required packages

```
npm install @react-navigation/native --save
npm install @react-navigation/stack --save
```

Create three Blank JavaScript files for three Screens as

```
Home.js
Login.js
Signup.js
```

Now open the App.js file and import the three screens into it along with `NavigationContainer` component from `@react-navigation/native` package and `createStackNavigator` component from `@react-navigation/stack` package.

Create a `Stack` object using `createStackNavigator` class and define the Navigation under `<NavigationContainer>` using `<Stack.Navigator>` component with `name`, `component` and `options` properties.

The component is the screen to which you want to navigate

```
//App.js
import * as React from 'react';
import { NavigationContainer } from '@react-navigation/native';
import { createStackNavigator } from '@react-navigation/stack';
import Home from './Home';
import Login from './Login';
import Signup from './Signup';

const Stack = createStackNavigator();

export default class App extends React.Component {
  render(){
    return (
      <NavigationContainer>
        <Stack.Navigator>
```

```

    <Stack.Screen
      name="Home"
      component={Home}
      options={{ title: 'Home Page' }}
    />
    <Stack.Screen
      name="Login"
      component={Login}
      options={{ title: 'Login Page' }}
    />
    <Stack.Screen
      name="Signup"
      component={Signup}
      options={{ title: 'Signup Page' }}
    />
  </Stack.Navigator>
</NavigationContainer>
);
}
}

```

Now create the sample Home Page with View, Text and Button. When we click on Login button, it should navigate to Login screen. Using `navigation.navigate()` function to navigate to the given `name` in App.js file.

```

//Home.js
import * as React from 'react';
import {View,Text,Button} from 'react-native';
export default class Home extends React.Component {
  constructor({navigation}){
    super(navigation);
  }
  render(){
    return (
      <View>
        <Text style={{color:'red',fontSize:24}}>Welcome to Home Page</Text>
        <Button
          title="Login"
          onPress={() =>
            this.props.navigation.navigate('Login')
          }
        />
      </View>
    );
  }
}

```

Now create the sample Login Page with View, Text, TextInput and Button. When we click on Login button, it should navigate to Login screen. Using `navigation.navigate()` function to navigate to the given `name` in App.js file.

```

//Login.js
import * as React from 'react';
import {View,Text,TextInput, Button} from 'react-native';
export default class Login extends React.Component {
  constructor({navigation}){
    super(navigation);
  }

```

```
}
render(){
  return (
    <View>
      <Text>Login</Text>
      <TextInput placeholder="User ID"/>
      <TextInput placeholder="Password" secureTextEntry/>
      <Button
        title="Login"
        onPress={() =>alert("Write the login logic here")}
      />
      <Button
        title="Signup"
        onPress={() =>
          this.props.navigation.navigate('Signup')
        }
      />
    </View>
  );
}
```

Now create the sample Signup Page with View, Text, TextInput and Button. Here we don't need navigation since we are not navigating from here for now.

```
//Signup.js
import * as React from 'react';
import {View,Text,TextInput, Button} from 'react-native';

export default class Signup extends React.Component {
  render(){
    return (
      <View>
        <Text>Login</Text>
        <TextInput placeholder="User ID"/>
        <TextInput placeholder="Password" secureTextEntry/>
        <Button
          title="Register"
          onPress={() =>alert('Registration logic goes here')}
        />
      </View>
    );
  }
}
```

Now run the project and test it

yarn start