To implement the "Remember Me" functionality in your login screen, you need to store user credentials securely using AsyncStorage in React Native. Here's how you can modify your `LoginScreen.tsx` to support this feature:

Steps to Implement "Remember Me":

1. Store the email and password in `AsyncStorage` when "Remember Me" is checked.
2. Retrieve stored credentials when the login screen loads.
3. Pre-fill the email and password fields if credentials exist.

Modify `LoginScreen.tsx`

Add the following updates to handle "Remember Me" functionality:

1. Install AsyncStorage

If you haven't installed `AsyncStorage`, add it first:

```
npm install @react-native-async-storage/async-storage
```

2. Update `LoginScreen.tsx`

Modify your `LoginScreen.tsx` file to include `AsyncStorage`:

```
import React, { useState, useEffect } from 'react';
import { View, Text, TextInput, TouchableOpacity, CheckBox } from 'react-native';
import AsyncStorage from '@react-native-async-storage/async-storage';

const LoginScreen = ({ navigation }) => {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [rememberMe, setRememberMe] = useState(false);

  useEffect(() => {
    // Load saved credentials on component mount
    const loadCredentials = async () => {
      const savedEmail = await AsyncStorage.getItem('savedEmail');
      const savedPassword = await AsyncStorage.getItem('savedPassword');
      if (savedEmail && savedPassword) {
        setEmail(savedEmail);
        setPassword(savedPassword);
        setRememberMe(true);
      }
    };
    loadCredentials();
  }, []);

  const handleLogin = async () => {
    // Perform login logic here

    if (rememberMe) {
      await AsyncStorage.setItem('savedEmail', email);
      await AsyncStorage.setItem('savedPassword', password);
    } else {
      await AsyncStorage.removeItem('savedEmail');
      await AsyncStorage.removeItem('savedPassword');
    }

    // Navigate to Home after login
    navigation.navigate('Home');
  };

  return (
    <View>
      <Text>Welcome back!</Text>
      <TextInput
        placeholder="Email"
        value={email}
        onChangeText={setEmail}
      />
      <TextInput
        placeholder="Password"
        value={password}
        onChangeText={setPassword}
        secureTextEntry
      />

      <View style={{ flexDirection: 'row', alignItems: 'center' }}>
        <CheckBox
          value={rememberMe}
          onValueChange={setRememberMe}
        />
        <Text>Remember Me</Text>
      </View>
      <TouchableOpacity onPress={handleLogin}>
        <Text>Login</Text>
      </TouchableOpacity>

      <TouchableOpacity onPress={() => navigation.navigate('Signup')}>
        <Text>New user? Sign Up</Text>
      </TouchableOpacity>
    </View>
  );
};

export default LoginScreen;
```

Explanation of Code Changes

1. **Retrieve Saved Credentials**
   - `useEffect` loads saved email and password from `AsyncStorage` when the screen loads.
   - If found, it pre-fills the input fields and enables "Remember Me."
2. **Saving Credentials on Login**
   - If "Remember Me" is checked, it saves the email and password.
   - If unchecked, it removes any previously saved credentials.
3. **Checkbox to Enable/Disable Feature**
   - Uses `CheckBox` to toggle "Remember Me" functionality.

Final Outcome

✅ If "Remember Me" is checked, credentials are saved and pre-filled on next login.
✅ If unchecked, credentials are cleared after logout or app restart.

Let me know if you need further modifications! 🚀