

React Native Notes

ASYNCSORAGE

The AsyncStorage JavaScript code is a facade that provides a clear JavaScript API, real Error objects, and non-multi functions. Each method in the API returns a Promise object. React Native AsyncStorage can be used to manage sessions.

If you want the user to log in once and don't want to log in again when the user opens the app after some time then, you have to store any variable in the app which can be checked and according to that, we will show the screen. For example, once we login we can store the user id in the AsyncStorage to remember the user and when we open the app again we check for the user id, If it is there in the AsyncStorage then we can directly send the user to the next screen without login else we will show the Login Screen. This process is called session management using AsyncStorage.

AsyncStorage can only be used to store small values in the form of a key and value pair because it has some limitations to store the data. You can store your value with respect to any key and then can access the value using the key again. It works perfectly on both platforms

Installation

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

To Import AsyncStorage in Code

```
import AsyncStorage from '@react-native-community/async-storage'
```

Store the value in AsyncStorage

```
AsyncStorage.setItem('any_key_here', this.state.textInputData);
```

React Native Notes

Get the value from the AsyncStorage

```
AsyncStorage.getItem('any_key_here')  
.then(value =>  
  //AsyncStorage returns a promise so adding a callback to get the value  
  this.setState({ getValue: value }) //Setting the value in Text );
```

Code

```
// import React in our code  
import React, {useState} from 'react';  
// import all the components we are going to use  
import { SafeAreaView, StyleSheet, View, TextInput, Text, TouchableOpacity, } from  
'react-native';  
// import AsyncStorage import AsyncStorage from '@react-native-community/  
async-storage';  
const App = () => {  
  // To get the value from the TextInput  
  const [textInputValue, setTextInputValue] = useState("");  
  // To set the value on Text  
  const [getValue, setGetValue] = useState("");  
  const saveValueFunction = () => {  
    // Function to save the value in AsyncStorage  
    if (textInputValue) {  
      // To check the input not empty  
      AsyncStorage.setItem('any_key_here', textInputValue);  
      // Setting a data to a AsyncStorage with respect to a key  
      setTextInputValue("");  
      // Resetting the TextInput  
      alert('Data Saved');  
      // Alert to confirm }  
    }  
  }  
}
```

React Native Notes

```
else { alert('Please fill data');
} };

const getValueFunction = () => {
// Function to get the value from AsyncStorage AsyncStorage.getItem('any_key_
here')
.then( (value) =>
// AsyncStorage returns a promise
// Adding a callback to get the value
setGetValue(value),
// Setting the value in Text );
};

return (
<SafeAreaView style={{flex: 1}}>
<View style={styles.container}>
<Text style={styles.titleText}></Text>
<TextInput placeholder="Enter Some Text here"
value={textInputValue}
onChangeText={(data) => setTextInputValue(data)}
underlineColorAndroid="transparent"
style={styles.textInputStyle} />
<TouchableOpacity
onPress={saveValueFunction}
style={styles.buttonStyle}>
<Text style={styles.buttonTextStyle}> SAVE VALUE </Text> </TouchableOpacity>
<TouchableOpacity onPress={getValueFunction} style={styles.buttonStyle}>
<Text style={styles.buttonTextStyle}> GET VALUE </Text> </TouchableOpacity>
<Text style={styles.textStyle}> {getValue} </Text>
</View>
</SafeAreaView>
); };
```

React Native Notes

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 10,
    backgroundColor:
    'white',
  },
  titleText: {
    fontSize: 22,
    fontWeight: 'bold',
    textAlign: 'center',
    paddingVertical: 20,
  },
  textStyle: {
    padding: 10,
    textAlign: 'center',
  },
  buttonStyle: {
    fontSize: 16,
    color: 'white',
    backgroundColor: 'green',
    padding: 5,
    marginTop: 32,
    minWidth: 250,
  },
  buttonTextStyle: {
    padding: 5,
    color: 'white',
    textAlign: 'center',
  },
});
```

React Native Notes

```
textInputStyle: {  
  textAlign: 'center',  
  height: 40,  
  width: '100%',  
  borderWidth: 1,  
  borderColor: 'green',  
},  
});  
  
export default App;
```