

React Native

• • •

Khalfani Bozeman

Mobile Browsing Stats in the United States

The distribution of mobile vs desktop internet usage in the United States over the past ten years reflects a similar pattern, where the percentage of web traffic on mobile has also surged. Despite the increase, mobile devices have yet to cement themselves as the preferred browsing option there.

As of August 2022, **53.74 percent** of the total web visits are currently mobile, compared to 46.26 percent coming from desktops.

Of the mobile internet browsers, Apple's Safari is proving to be the most popular choice among all mobile internet users in the US. The latest figures show that **52.84 percent** of all US mobile users use Safari to browse the internet, with Google Chrome coming in second with 41.34 percent.

Who is using React Native?

Thousands of apps are using React Native, from established Fortune 500 companies to hot new startups. If you are curious to see what can be accomplished with React Native, check out these apps!



React Native is shaping mobile, web, and desktop experiences within Meta's product ecosystem, from Facebook Marketplace, Messenger Desktop, Ads Manager to the Oculus companion app and many more.



Facebook
Android · iOS



Facebook Ads Manager
Android · iOS



Oculus
Android · iOS



Microsoft leverages the power of React Native to deliver excellent customer experiences in some of its most well known apps. Microsoft doesn't stop at mobile platforms either -- Microsoft leverages React Native to target desktop too! Find out more in the [dedicated showcase](#) for React Native Windows and macOS.



Microsoft Office
Android · iOS



Microsoft Outlook
Android · iOS



Microsoft Teams
Android · iOS



Xbox Game Pass
Android · iOS



Skype
Android · iOS

<https://reactnative.dev/showcase>

Largest Companies That Use React: At a Glance

Company	Revenue	Number of employees
Meta	\$85.96 billion (2020)	68,177 (2021)
Netflix	\$29.69 billion (2021)	12,135 (2021)
Uber	\$11.1 billion (2020)	24,700 (2021)
Yahoo!	\$5.16 billion (2021)	8,500 (2021)
Airbnb	\$3.38 billion (2020)	5,597 (2020)
Atlassian	\$2.08 billion (2021)	6,433 (2021)
Dropbox	\$1.91 billion (2020)	2,548 (2021)
The New York Times	\$1.78 billion (2020)	4,700 (2020)
Zendesk	\$1.03 billion (2020)	4,000+ (2021)
Asana	\$227 million (2021)	1080 (2021)

<https://careerkarma.com/blog/companies-that-use-react/>

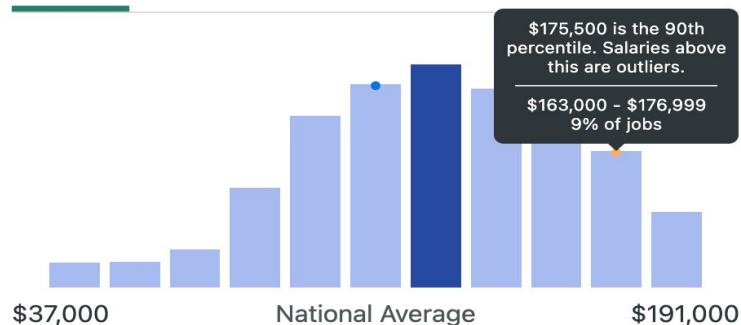


React Native

REACT NATIVE

Salary Jobs

Yearly Monthly Weekly Hourly Table View



\$131,388 /year
\$63 /hour

React Native Salary Comparison by Location

Nationwide
United States

\$131,388

<https://www.ziprecruiter.com/Salaries/React-Native-Salary>

React developer salaries by location: Top 5 highest-paying cities

US City	Salary
San Mateo, CA	\$141,087
Juneau, AK	\$138,387
Boston, MA	\$138,127
Santa Monica, CA	\$136,390
Quincy, MA	\$136,383 [5]

<https://www.coursera.org/articles/react-developer-salary>

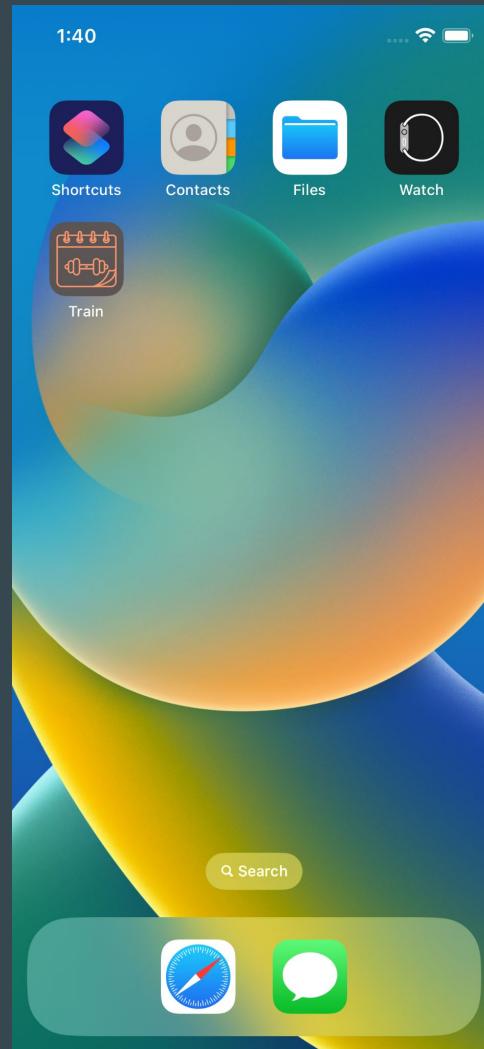
100 *SECONDS OF*

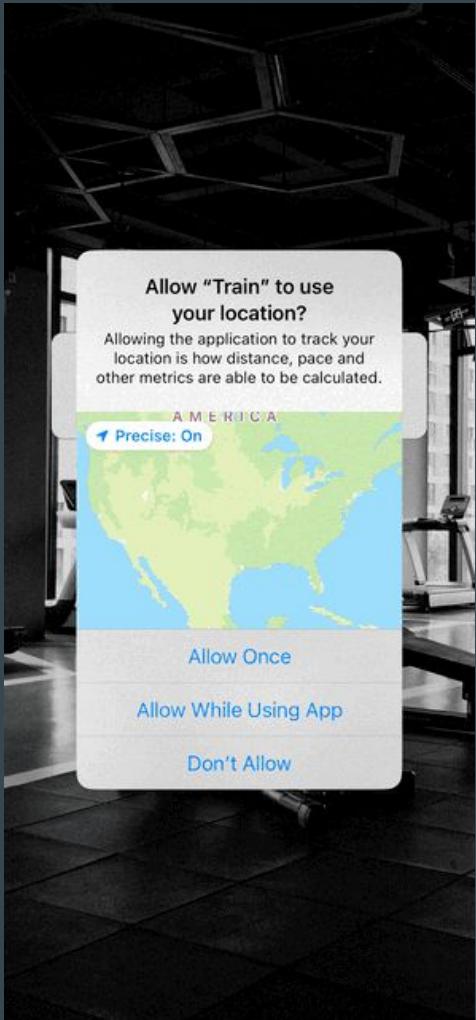


<https://www.youtube.com/c/Fireship>

“Train”

<https://testflight.apple.com/join/SPvPRLy0>





1:36

< Back Home

 COMING SOON

Local Leagues
Create a local sports league to compete with or against friends.

 COMING SOON

The Board

Calendar icon, Pencil icon, Home icon, Trophy icon, Music icon

1:33

< Back Home

 COMING SOON

Local Leagues
Create a local sports league to compete with or against friends.

 COMING SOON

The Board

Calendar icon, Pencil icon, Home icon, Trophy icon, Music icon

Presentation Topics

- Environment Setup / React Native CLI
- Navigation API (*Mobile App Development for iOS and Android, Ch. 4 / Ch. 10*)
 - AsyncStorage API (Ch. 5 / Ch. 11)
 - Geolocation API (Ch. 7 / Ch. 13)
- *Platform API (Ch. 8 / 14) / PermissionsAndroid API (Ch. 7)*
 - Maps (Ch. 7 / Chapter 13)
- Background Timer & Background Processing (Ch. 2)
 - Fetch API
 - Alert API (Ch. 4)

Environment Setup

Expo Go Quickstart

React Native CLI Quickstart

Follow these instructions if you need to build native code in your project. For example, if you are integrating React Native into an existing application, or if you ran "prebuild" from Expo to generate your project's native code, you'll need this section.

The instructions are a bit different depending on your development operating system, and whether you want to start developing for iOS or Android. If you want to develop for both Android and iOS, that's fine - you can pick one to start with, since the setup is a bit different.

Development OS #

macOS

Windows

Linux

Target OS

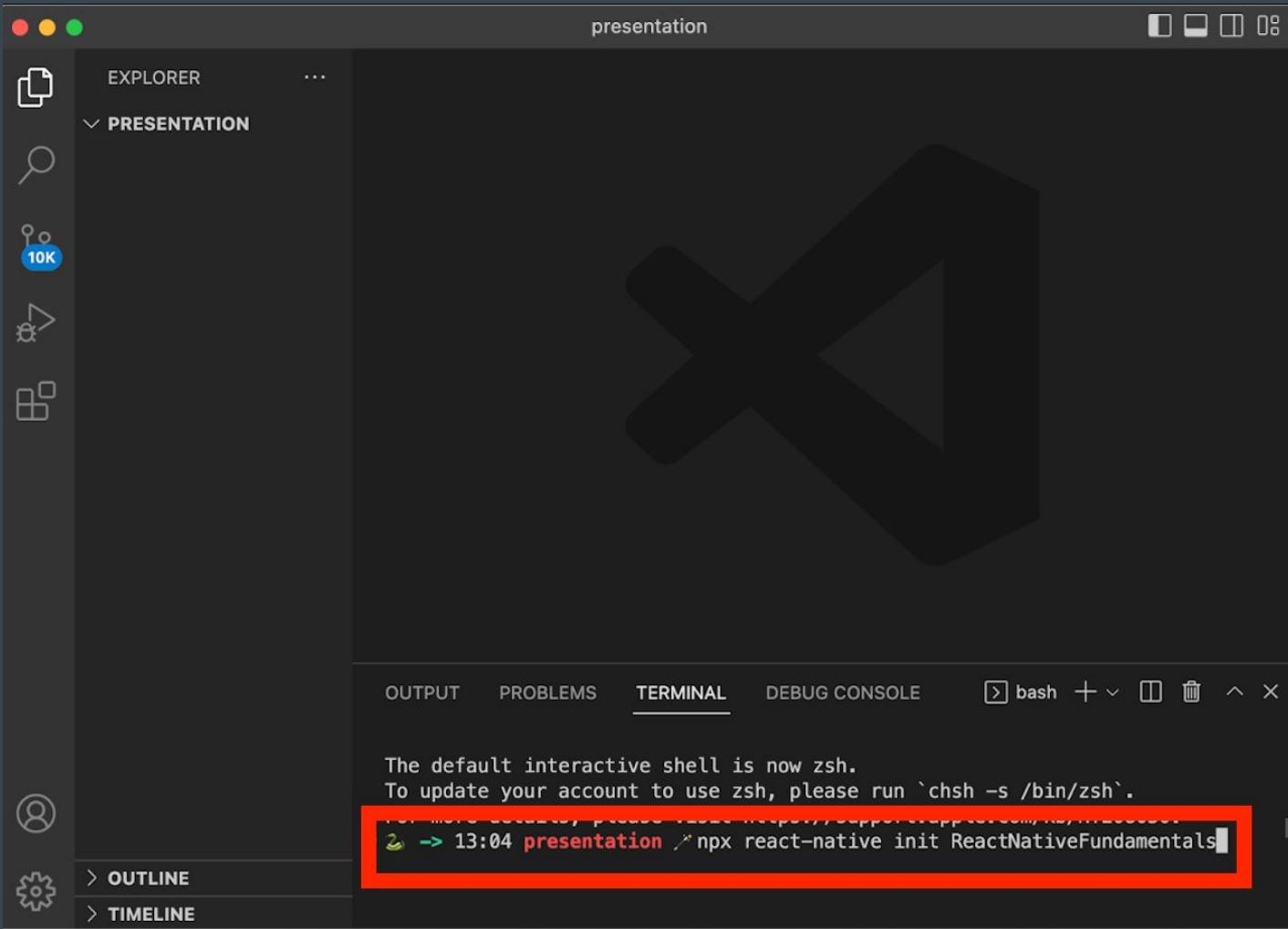
Android

iOS

Installing dependencies

You will need Node, Watchman, the React Native command line interface, a Ruby version manager, Xcode, and CocoaPods.

<https://reactnative.dev/docs/environment-setup>



Generated File Structure

AndroidManifest.xml — ReactNativeFundamentals

EXPLORER ...

AndroidManifest.xml ×

android > app > src > main > AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.reactnativefundamentals">

    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:name=".MainApplication"
        android:label="@string/app_name"
        android:icon="@mipmap/ic_launcher"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:allowBackup="false"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:configChanges="keyboard|keyboardHidden|orientation|screenSize"
            android:launchMode="singleTask"
            android:windowSoftInputMode="adjustResize"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

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

REACTNATIVEFUNDAMENTALS

__tests__
_bundle
.idea
.vscode
android
.gradle
app
src
debug
main
java
jni
res
AndroidManifest.xml
_BUCK
build_defs.bzl
build.gradle
debug.keystore

OUTLINE
TIMELINE
XML DOCUMENT
JAVA PROJECTS

master ⌂ 0 △ 0 ⌂ React Native Packager ⌂ Ln 1, Col 1 Spaces: 2 UTF-8 LF XML ⌂ Prettier ⌂ ⌂ ⌂

The screenshot shows the VS Code interface with the following details:

- EXPLORER** view: Shows the project structure under "REACT...". A red box highlights the "ios" folder and its contents: "build/generate...", "Pods", "ReactNativeFundamentals", "Images.xcass...", "AppDelegate.h", "AppDelegate....", "Info.plist", "LaunchScreen...", "main.m", and three "ReactNativeFundamentals" folders.
- EDITOR** tab: Displays the "Info.plist" file content.
- STATUS BAR**: Shows "master", "React Native Packager", "Ln 1, Col 1", "Tab Size: 4", "UTF-8", "LF", "XML", "Prettier", and icons for search, file operations, and notifications.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>CFBundleDevelopmentRegion</key>
    <string>en</string>
    <key>CFBundleDisplayName</key>
    <string>ReactNativeFundamentals</string>
    <key>CFBundleExecutable</key>
    <string>$EXECUTABLE_NAME</string>
    <key>CFBundleIdentifier</key>
    <string>$(PRODUCT_BUNDLE_IDENTIFIER)</string>
    <key>CFBundleInfoDictionaryVersion</key>
    <string>6.0</string>
    <key>CFBundleName</key>
    <string>$(PRODUCT_NAME)</string>
    <key>CFBundlePackageType</key>
    <string>APPL</string>
    <key>CFBundleShortVersionString</key>
    <string>1.0</string>
    <key>CFBundleSignature</key>
    <string>????</string>
    <key>CFBundleVersion</key>
    <string>1</string>
    <key>LSRequiresiPhoneOS</key>
    <true/>
    <key>NSAppTransportSecurity</key>
```

App.js — ReactNativeFundamentals

EXPLORER ... App.js App.js

REACT... .idea .vscode android ios node_modules _node-version _ruby-version .buckconfig .eslintrc.js .flowconfig .gitignore .prettierrc.js .watchmanconfig App.js app.json babel.config.js Gemfile index.js metro.config.js package-lock.json package.json

66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95

```
<StatusBar  
    barStyle={isDarkMode ? 'light-content' : 'dark-content'}  
    backgroundColor={backgroundStyle.backgroundColor}  
/>  
<ScrollView  
    contentInsetAdjustmentBehavior="automatic"  
    style={backgroundStyle}>  
    <Header />  
    <View  
        style={  
            | backgroundColor: isDarkMode ? Colors.black : Colo  
        }|>  
    <Section title="Step One">  
        | Edit <Text style={styles.highlight}>App.js</Text>  
        | screen and then come back to see your edits.  
    </Section>  
    <Section title="See Your Changes">  
        | <ReloadInstructions />  
    </Section>  
    <Section title="Debug">  
        | <DebugInstructions />  
    </Section>  
    <Section title="Learn More">  
        | Read the docs to discover what to do next:  
    </Section>  
        | <LearnMoreLinks />  
    </View>  
    </ScrollView>  
    </SafeAreaView>
```

master 0 △ 0 React Native Packager Spaces: 2 UTF-8 LF JavaScript Prettier

Running The Application

The screenshot displays a React Native development setup. On the left, the **EXPLORER** panel shows the file structure with `index.js` selected. The **EDITOR** panel shows the code for `index.js`, which includes imports for `AppRegistry` and `App`. The **TERMINAL** panel on the right shows the Metro Bundler interface with the message "Welcome to Metro v0.72.3" and instructions for reloading and opening developer menus. It also shows the command `BUNDLE ./index.js` and a log message indicating the app is running. A red box highlights the terminal window. At the bottom, the status bar shows the current branch is `master` and the file is `index.js`.

iPhone 14
iOS 16.0

EXPLORER index.js

index.js

1:51

Welcome to React Native

Step One

Edit `App.js` to change this screen and then come back to see your edits.

See Your Changes

Press **Cmd + R** in the simulator to reload your app's code.

Debug

Press **Cmd + D** in the simulator or Shake your device to open the React Native debug menu.

Learn More

Read the docs to discover what to do next:

The Basics Explains a Hello World for

fault interactive shell is now zsh.
ate your account to use zsh, please run `chsh -s /bin/zsh`.
re details, please visit <https://support.apple.com/kb/HT208050>.

13:32 **ReactNativeFundamentals** \$ `./npx react-native run-ios --simulator="iPhone 14"`

bund Xcode workspace "ReactNativeFundamentals.xcworkspace".

aunching iPhone 14 (iOS 16.0)

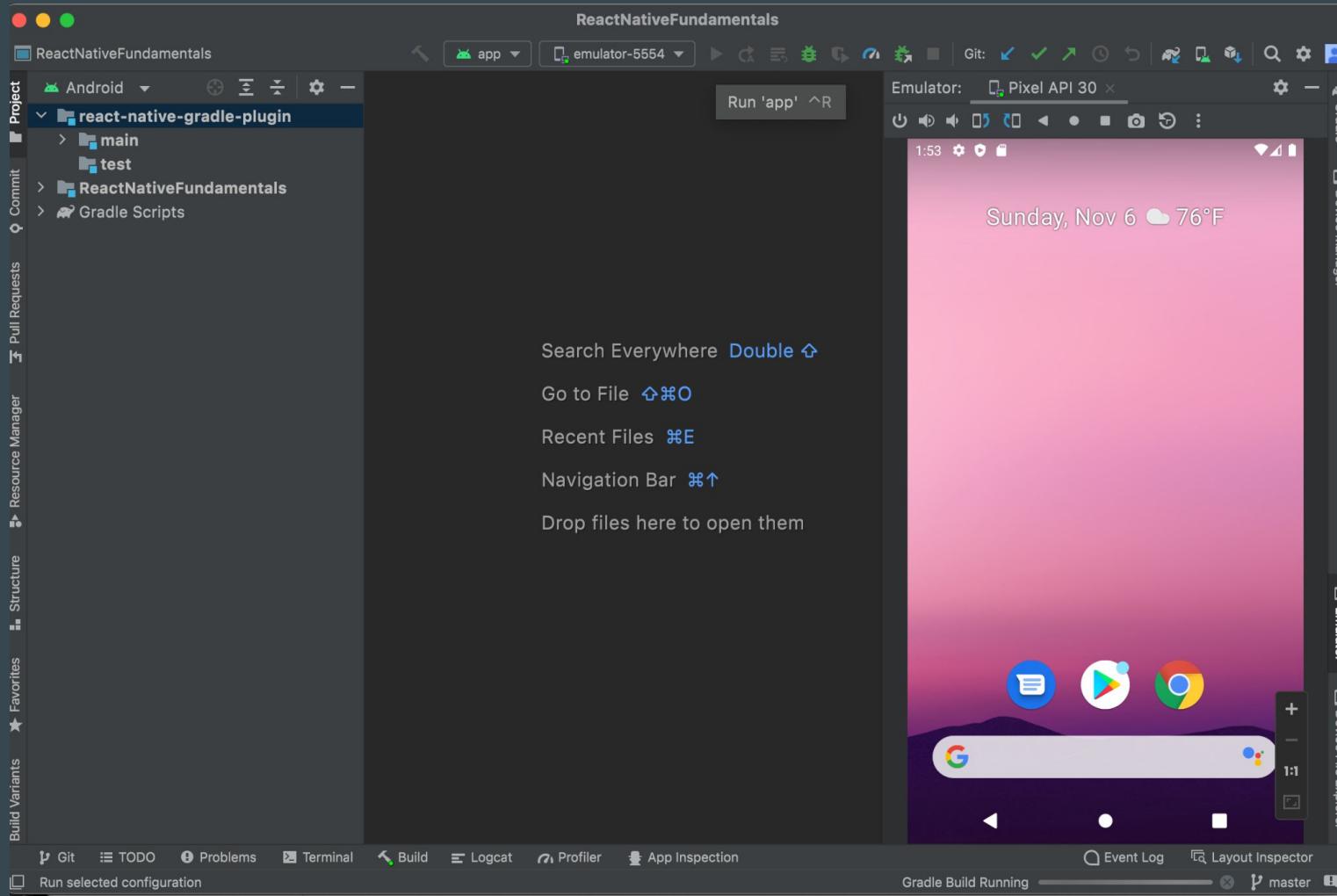
uilding (using "xcodebuild -workspace ReactNativeFundamentals.xcworkspace -configuration D

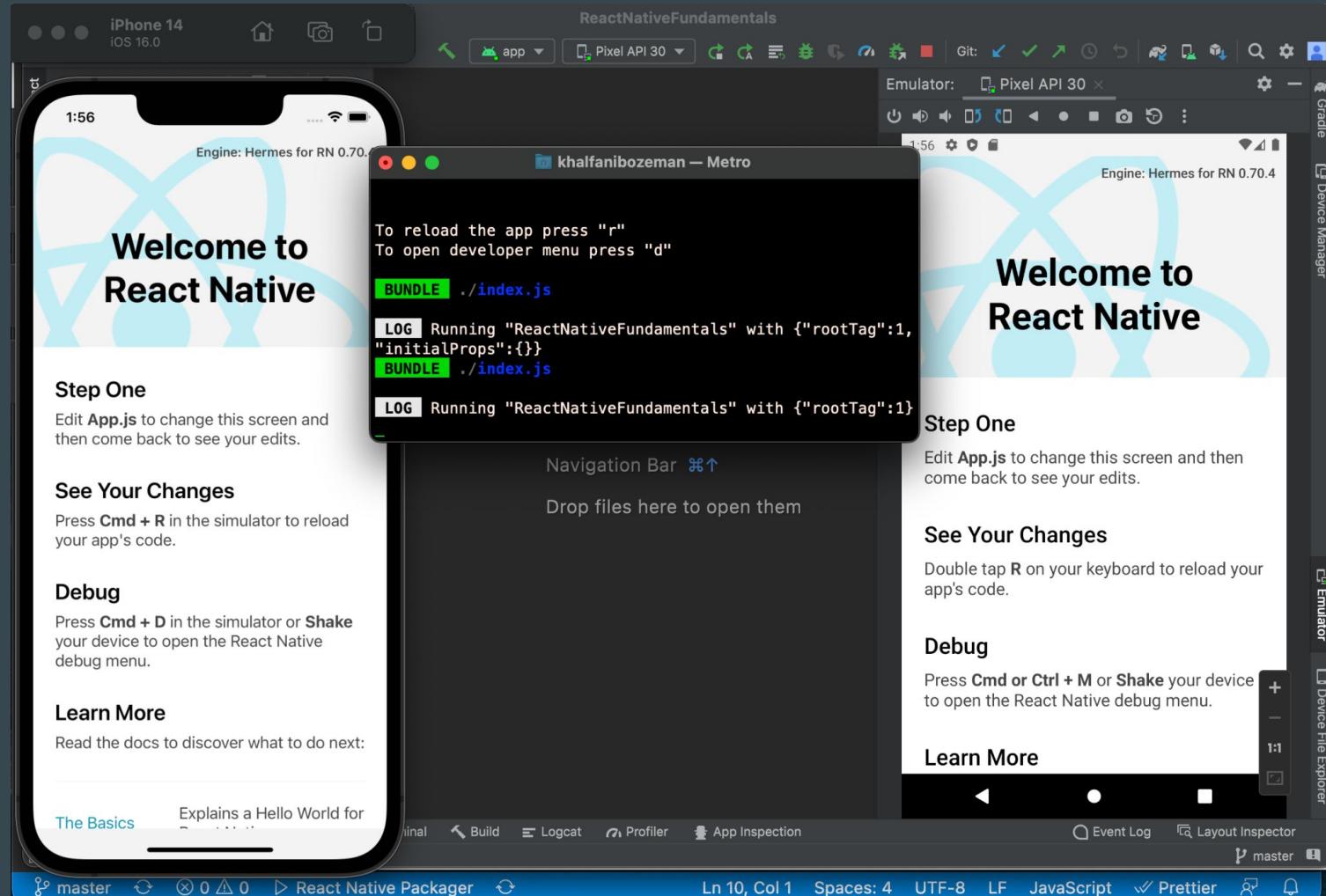
scheme ReactNativeFundamentals -destination id=B34AF5C8-51D6-4373-8D54-39EFAFE74C1D")

ss Successfully built the app

master index.js React Native Packager

Ln 10, Col 1 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier ⚡ 🔍





Let's code!

(<https://github.com/egrep6021ad/ReactNativeFundamentals>)

The screenshot shows a dark-themed instance of Visual Studio Code (VS Code) with the following interface elements:

- Top Bar:** Shows the file name "index.js — ReactNativeFundamentals" and standard window control buttons.
- Left Sidebar (Explorer):** Displays the project structure:
 - A tree view under "REACT..." showing ".idea", ".vscode", "android", "ios", "node_modules" (with sub-folders like "_node-version", "_ruby-version", ".buckconfig", ".eslintrc.js", ".flowconfig", ".gitignore", ".prettierrc.js", ".watchmanconfig"), "App.js", "app.json", "babel.config.js", "Gemfile", and "index.js".
 - Icons for users, timeline, and Java projects.
- Central Area (Editor):** The "index.js" file is open, showing the following code:

```
1  /**
2  * @format
3  */
4
5  import {AppRegistry} from 'react-native';
6  import App from './App';
7  import {name as appName} from './app.json';
8
9  AppRegistry.registerComponent(appName, () => App);
10
```
- Bottom Navigation Bar:** Includes tabs for "OUTPUT", "PROBLEMS", "TERMINAL" (which is selected), and "DEBUG CONSOLE".
- Terminal Output:** Shows the Xcode build log:

```
generation)
2022-11-06 13:47:10.277 xcodebuild[18403:293254] Requested but did not find extension point with
identifier Xcode.InterfaceBuilderBuildSupport.PlatformDefinition
info Installing "/Users/khalfanibozeman/Library/Developer/Xcode/DerivedData/ReactNativeFundamentals-ccqo0tdyiuuhcrbrddfbmqpuwkyb/Build/Products/Debug-iphonesimulator/ReactNativeFundamentals.ap
p"
info Launching "org.reactjs.native.example.ReactNativeFundamentals"
success Successfully launched the app on the simulator
○ 13:47 ReactNativeFundamentals
```
- Bottom Status Bar:** Shows "master", "React Native Packager", "Ln 10, Col 1", "Spaces: 4", "UTF-8", "LF", "JavaScript", "Prettier", and icons for search, refresh, and notifications.

App.js — ReactNativeFundamentals

iPhone 14
iOS 16.0

EXPLORER ... App.js X

REACTNATIVEFUNDAME... App.js

.idea
.vscode
android
ios
node_modules
 _node-version
 _ruby-version
.buckconfig
.eslintrc.js
.flowconfig
.gitignore
.prettierrc.js
watchmanconfig

App.js app.json
babel.config.js
Gemfile
index.js
metro.config.js
package-lock.json
package.json

> OUTLINE
> TIMELINE
> JAVA PROJECTS

56
57 const App: () => Node = () => {
58 const isDarkMode = useColorScheme() === 'dark';
59
60 const backgroundStyle = {
61 backgroundColor: isDarkMode ? Colors.darker : Colors.lighter,
62 };
63
64 return (
65 <StatusBar
66 barStyle={isDarkMode ? 'light-content' : 'dark-content'}
67 backgroundColor={backgroundStyle.backgroundColor}
68 />
69 <ScrollView
70 contentInsetAdjustmentBehavior="automatic"
71 style={backgroundStyle}>
72 <Header />
73 <View
74 style={{
75 backgroundColor: isDarkMode ? Colors.black : Colors.white
76 }}>
77 <Section title="Step One">
78 Edit <Text style={styles.highlight}>App.js</Text> to cha
79 screen and then come back to see your edits.
80 </Section>
81 <Section title="See Your Changes">
82 <ReloadInstructions />
83 </Section>
84 </View>

2:02

Welcome to
React Native

Step One
Edit App.js to change this screen and
then come back to see your edits.

See Your Changes
Press Cmd + R in the simulator to reload
your app's code.

Debug
Press Cmd + D in the simulator or Shake
your device to open the React Native
debug menu.

Learn More
Read the docs to discover what to do next:

The Basics Explains a Hello World for

Ln 83, Col 35 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier

The screenshot shows a developer's environment with several open windows:

- Terminal:** Displays an **ERROR** message: `[Error: TransformError SyntaxError: /Users/khalfan.../ReactNativeFundamentals/App.js: Unexpected token, expected "," (21:4)]`. A red box highlights the error message and the problematic code block.
- Code Editor:** Shows the `App.js` file. The problematic code block is highlighted with a red box:

```
19 |   mainContainer: {  
20 |     display: 'flex'  
21 |     justifyContent: 'center',  
22 |     ^  
23 |     alignItems: 'center',  
24 |     backgroundColor: 'yellow'  
25 |   }  
26 |  
27 |   const styles = StyleSheet.create({  
28 |     mainContainer: {  
29 |       display: 'flex'  
30 |       justifyContent: 'center',  
31 |       alignItems: 'center',  
32 |       backgroundColor: 'yellow'  
33 |     }  
34 |   });  
35 |  
36 |   export default App;
```
- iPhone 14 Simulator:** Shows a red screen with the message "Failed to compile". Below it, a "Syntax Error" message is displayed: "Unexpected token, expected ","". The source code for `App.js` is shown, with line 21:5 highlighted. The status bar indicates the time is 2:08 and the device is running iOS 16.0.
- File Browser:** Shows the project structure with files like `_ruby-version`, `.buckconfig`, `.eslintrc.js`, `.flowconfig`, `.gitignore`, `.prettierrc.js`, `.watchmanconfig`, `App.js`, `app.json`, `babel.config.js`, `Gemfile`, `index.js`, `metro.config.js`, `package-lock.json`, and `package.json`.

The bottom of the screen displays various developer tools and status indicators, including "React Native Packager", "Ln 9, Col 32", "Spaces: 2", "UTF-8 LF JavaScript", and "Prettier".

khalfanibozeman — Metro

```
in RCTView (created by View)
in View (created by AppContainer)
in RCTView (created by View)
in View (created by AppContainer)
in AppContainer
in ReactNativeFundamentals(RootElement), js engine: hermes
info Reloading app...
BUNDLE ./index.js

LOG Running "ReactNativeFundamentals" with {"rootTag":31,"initialProps":{}}


  _node-version
  _ruby-version
  .buckconfig
  .eslintrc.js
  .flowconfig
  .gitignore
  .prettierrc.js
  .watchmanconfig
  App.js M
  app.json
  babel.config.js
  Gemfile
  index.js
  metro.config.js
  package-lock.json
  package.json

  > OUTLINE
  > TIMELINE
  > JAVA PROJECTS
```

iPhone 14
iOS 16.0

2:10

```
from 'react';
ode} from 'react';
eaView, StyleSheet, Text, View} from
=> Node = () => {
iew>
  <View style={styles.mainContainer}>
    <Text> Hello Class! </Text>
  </View>
</SafeAreaView>
);
};

const styles = StyleSheet.create({
  mainContainer: {
    display: 'flex',
    justifyContent: 'center',
    alignItems: 'center',
    height: '100%',
    width: '100%',
    backgroundColor: 'yellow',
  },
});
export default App;
```

Hello Class!

master* ↻ ⊗ 0 △ 0 ▷ React Native Packager ↻ Ln 21, Col 19 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂

Navigation API

(<https://reactnavigation.org/docs/native-stack-navigator/>)



Guides

API Reference



NavigationContainer

ServerContainer

Group

Screen

Options for screens

Route prop

Navigation prop

NavigationContext

Navigation events

Navigation state

Link

Developer tools

Elements Library

Navigators



Version: 6.x

NavigationContainer

The `NavigationContainer` is responsible for managing your app state and linking your top-level navigator to the app environment.

The container takes care of platform specific integration and provides various useful functionality:

1. Deep link integration with the `linking` prop.
2. Notify state changes for `screen tracking`, `state persistence` etc.
3. Handle system back button on Android by using the `BackHandler` API from React Native.

Usage:

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

const Stack = createStackNavigator();
```

Ref

Methods on the ref

Props

`initialState`
`onStateChange`
`onReady`
`onUnhandledAction`
`linking`
`fallback`
`documentTitle`
`theme`
`independent`

package.json — ReactNativeFundamentals

EXPLORER ... package.json M X

REACTNATIVEFUNDAMENTALS android ios node_modules Screens Home.js _node-version _ruby-version .buckconfig .eslintrc.js .flowconfig .gitignore .prettierrc.js .watchmanconfig App.js app.json babel.config.js Gemfile index.js metro.config.js package-lo... package.json M

8 "ios": "react-native run-ios",
9 "start": "react-native start",
10 "test": "jest",
11 "lint": "eslint ."

12 "dependencies": {
13 "@react-navigation/native-stack": "^6.9.1",
14 "react": "18.1.0",
15 "react-native": "0.70.4"
16 },

17 "devDependencies": {
18 "@babel/core": "^7.12.9",
19 "@babel/runtime": "^7.12.5",
20 "@react-native-community/eslint-config": "^2.0.0",
21 "babel-jest": "^26.6.3",
22 "eslint": "^7.32.0",
23 "jest": "^26.6.3",
24 "metro-react-native-babel-preset": "0.72.3",
25 "react-test-renderer": "18.1.0"
26 },
27

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE bash + ×

14:20 ReactNativeFundamentals ↵ npm install @react-navigation/native-stack

added 18 packages, and audited 1311 packages in 9s
106 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities

14:21 ReactNativeFundamentals ↵

master* 0 △ 0 ▶ React Native Packager

Ln 1, Col 1 Spaces: 2 UTF-8 LF JSON Prettier

package.json — ReactNativeFundamentals

Highlights and Notes ... package.json M X

REACTNATIVEFUNDAMENTALS

- > android
- > ios
- > node_modules
- > .bin
- > @ampproject
- > @babel
- > @bcoe
- > @cnakazawa
- > @eslint
- > @hapi
- > @humanwhocodes
- > @istanbuljs
- > @jest
- > @jridgewell
- > @react-native
- > @react-native-community
- > @react-navigation
- > @sideway
- > @sinclair
- > @sinonjs
- > @toottallnate

4

package.json > {} dependencies

```
8 "start": "react-native start",
9 "test": "jest",
10 "lint": "eslint ."
11 },
12 "dependencies": {
13   "@react-navigation/native": "^6.0.13",
14   "@react-navigation/native-stack": "^6.9.1",
15   "react": "18.1.0",
16   "react-native": "0.70.4"
17 },
18 "devDependencies": {
19   "@babel/core": "^7.12.9",
20   "@babel/runtime": "^7.12.5",
21   "@react-native-community/eslint-config": "^2.0.0",
22   "babel-jest": "^26.6.3",
23   "eslint": "^7.32.0",
24   "jest": "^26.6.3",
25   "metro-react-native-babel-preset": "0.72.3",
}
```

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

bash + ↻ □ ✎ ^ ×

```
found 0 vulnerabilities
● 14:34 ReactNativeFundamentals ➔ npm install @react-navigation/native
```

up to date, audited 1311 packages in 2s

106 packages are looking for funding
run `npm fund` for details

found 0 vulnerabilities

```
○ 14:34 ReactNativeFundamentals ➔
```

master* ↻ ⚡ 0 ⚡ 0 ▷ React Native Packager ↻

Ln 17, Col 1 Spaces: 2 UTF-8 LF { JSON ✓ Prettier 🔍 🔍

package.json — ReactNativeFundamentals

EXPLORER ... package.json M X

REACTNATIVEFUNDAMENTALS

- > android
- > ios
- > node_modules
- Screens ●
 - Home.js
 - _node-version
 - _ruby-version
 - .buckconfig
 - .eslintrc.js
 - .flowconfig
 - .gitignore
 - .prettierrc.js
 - .watchmanconfig
 - App.js M
 - app.json
 - babel.config.js
 - Gemfile
 - index.js
 - metro.config.js
 - package-lock.json M
 - package.json M

4

package.json > {} dependencies > @react-navigation/native-stack

```
7   "ios": "react-native run-ios",
8   "start": "react-native start",
9   "test": "jest",
10  "lint": "eslint ."

11 }
12 "dependencies": {
13   "@react-navigation/native": "^6.0.13",
14   "@react-navigation/native-stack": "^6.9.1",
15   "react": "18.1.0",
16   "react-native": "0.70.4",
17   "react-native-safe-area-context": "^4.4.1",
18   "react-native-screens": "^3.18.2"
19 }

20 "devDependencies": {
21   "@babel/core": "^7.12.9",
22   "@babel/runtime": "^7.12.5",
23   "@react-native-community/eslint-config": "^2.0.0",
24   "babel-jest": "^26.6.3",
25   "eslint": "^7.32.0",

```

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

bash + □ ✖ ^ ×

● 14:48 ReactNativeFundamentals ↗ npm install react-native-screens react-native-safe-area-context

up to date, audited 1511 packages in 2s

106 packages are looking for funding
run `npm fund` for details

found 0 vulnerabilities

○ 14:48 ReactNativeFundamentals ↗

master* ⏪ 0 △ 0 ▷ React Native Packager ⏪ Ln 14, Col 35 Spaces: 2 UTF-8 LF { JSON ✓ Prettier 🔍 🔍

Home.js — ReactNativeFundamentals

EXPLORER ... js Home.js U

Screens > js Home.js

```
1 import React from 'react';
2 import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
3
4 export default function Home() {
5   return (
6     <SafeAreaView>
7       <View style={styles.mainContainer}>
8         <Text> Hello Class! </Text>
9       </View>
10    </SafeAreaView>
11  );
12}
13const styles = StyleSheet.create({
14  mainContainer: [
15    display: 'flex',
16    justifyContent: 'center',
17    alignItems: 'center',
18    height: '100%',
19    width: '100%',
20    backgroundColor: 'yellow',
21  ],
22});
23
```

_node-version
_ruby-version
.buckconfig
.eslintrc.js
.flowconfig
.gitignore
.prettierrc.js
.watchmanconfig
App.js M
{} app.json
babel.config.js
Gemfile
index.js

OUTLINE
TIMELINE
JAVA PROJECTS

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

bash + □ × ^ ×

als-ccqootdyiiuhcrbrddfbmqpuwkyb/Build/Products/Debug-iphonesimulator/ReactNativeFundamentals.ap
p"
info Launching "org.reactjs.native.example.ReactNativeFundamentals"
success Successfully launched the app on the simulator
14:25 ReactNativeFundamentals ↻

master* ↵ ⌂ ⌂ 0 △ 0 ▷ React Native Packager ↵ Ln 17, Col 26 Spaces: 2 UTF-8 LF JavaScript ✅ Prettier ⌂ ⌂

App.js — ReactNativeFundamentals

EXPLORER ...

REACTNATIVEFUNDAME... ▾

- > android
- > ios
- > node_modules
- ↳ Screens ●
 - ↳ Home.js U
 - ↳ _node-version
 - ↳ _ruby-version
 - ↳ .buckconfig
 - ↳ .eslintrc.js
 - ↳ .flowconfig
 - ↳ .gitignore
 - ↳ .prettierrc.js
 - ↳ .watchmanconfig
- ↳ App.js M
- {.} app.json
- ↳ babel.config.js
- ↳ Gemfile
- ↳ index.js
- ↳ metro.config.js
- ↳ package-lo... M
- ↳ package.json M

OUTLINE

TIMELINE

JAVA PROJECTS

App.js

```
1 import React from 'react';
2 import type {Node} from 'react';
3 import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
4 import {NavigationContainer} from '@react-navigation/native';
5 import {createNativeStackNavigator} from '@react-navigation/native-stack';
6 import Home from './Screens/Home';

7
8 const App: () => Node = () => {
9   const Stack = createNativeStackNavigator();
10  return (
11    <NavigationContainer>
12      <Stack.Navigator initialRouteName="Home">
13        <Stack.Screen name="Home" component={Home} />
14      </Stack.Navigator>
15    </NavigationContainer>
16  );
17}
18 export default App;
19
```

master* ↻ ✘ 0 △ 0 ▷ React Native Packager ↻

Ln 16, Col 5 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier 🔍 🔔

The screenshot shows a development environment with three main panes:

- Left Pane:** A terminal window titled "khalfanibozeman — Metro" displaying a stack trace of component creation. The stack trace starts with "in NativeStackNavigator (created by App)" and continues through various RN components like EnsureSingleNavigator, BaseNavigationContainer, ThemeProvider, NavigationContainerInner, and App, ending with "in ReactNativeFundamentals(RootComponent), js engine: hermes".
- Middle Pane:** An IDE editor showing the file "index.js" from the "ReactNativeFundamentals" project. The code contains a `<Text>` component with the text "Hello Class!". It also includes a `StyleSheet.create` block defining a `mainContainer` style with flex properties and a yellow background color.
- Right Pane:** A mobile application interface for an iPhone 14 running iOS 16.0. The screen displays a red "Render Error" box with the message: "requireNativeComponent: 'RNSScreenStackHeaderConfig' was not found in the UIManager." Below this is a "Source" section showing the problematic code from the "getNativeComponentAttributes.js" file at line 27, and a "Call Stack" section listing the stack trace from "getNativeComponentAttributes" down to "runApplication".

At the bottom of the interface, there are navigation icons for "master*", "React Native Packager", and "Prettier". The status bar indicates the current line (Ln 2, Col 39), spaces (Spaces: 2), encoding (UTF-8), and file type (JavaScript).

Home.js — ReactNativeFundamentals

EXPLORER ⋮

REACTNATIVEFUNDAMENTALS

- > android
- > ios
- > node_modules
- ↳ Screens ●
 - ↳ Home.js U
- ↳ _node-version
- ↳ _ruby-version
- ↳ .buckconfig
- ↳ .eslintrc.js
- ↳ .flowconfig
- ↳ .gitignore
- ↳ .prettierrc.js
- ↳ .watchmanconfig
- ↳ App.js M
- ↳ app.json
- ↳ babel.config.js
- ↳ Gemfile
- ↳ index.js
- ↳ metro.config.js
- ↳ package-lock.json M
- ↳ package.json M

OUTLINE ⋮

TIMELINE ⋮

JAVA PROJECTS ⋮

js Home.js U X

Screens > js Home.js

```
1 import React from 'react';
2 import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
3
4 export default function Home() {
5   return [
6     <SafeAreaView>
7       <View style={styles.mainContainer}>
8         | <Text> Hello Class! </Text>
9       </View>
10    </SafeAreaView>
11  ];
12}
```

TERMINAL

```
14:42 ReactNativeFundamentals ✘ cd ios
14:42 ios ✘ pod install
ignoring ffi-1.15.4 because its extensions are not built. Try: gem pristine ffi --version 1.15.4
[Codegen] Generating ./build/generated/ios/React-Codegen.podspec.json
Analyzing dependencies
[Codegen] Found FBReactNativeSpec
Downloading dependencies
Generating Pods project
Setting REACT_NATIVE build settings
Setting CLANG_CXX_LANGUAGE_STANDARD to c++17 on /Users/khalfanibozeman/Desktop/Fall22/Mobile-App-Development/presentation/ReactNativeFundamentals/ios/ReactNativeFundamentals.xcodeproj
Pod install took 17 [s] to run
Integrating client project
Pod installation complete! There are 59 dependencies from the Podfile and 49 total pods installed.
14:43 ios ✘ cd ..
14:43 ReactNativeFundamentals ✘
```

Ln 9, Col 14 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂

The screenshot shows a developer environment with several components:

- Top Left:** A code editor window titled "App.js - ReactNativeFundamentals" containing the following code:

```
op.js M X
op.js

import React from 'react';
import type {Node} from 'react';
import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
import {NavigationContainer} from '@react-navigation/native';
import {createNativeStackNavigator} from '@react-navigation/native-stack';
import Home from './Screens/Home';

const Stack = createNativeStackNavigator();
const App: () => Node = () => {
  return (
    <NavigationContainer>
      <Stack.Navigator initialRouteName="Home">
        <Stack.Screen name="Home" component={Home}>
        </Stack.Screen>
      </Stack.Navigator>
    </NavigationContainer>
  );
};
export default App;
```

- Top Center:** An emulator window titled "Pixel API 30" showing a yellow screen with the text "Hello Class!".
- Top Right:** An emulator window titled "iPhone 14" showing a yellow screen with the text "Home".
- Bottom Center:** A terminal window titled "khalfanibozeman — Metro" displaying build logs:

```
To reload the app press "r"
To open developer menu press "d"
BUNDLE ./index.js
LOG  Running "ReactNativeFundamentals" with {"rootTag":1,"initialProps":{}}
BUNDLE ./index.js
LOG  Running "ReactNativeFundamentals" with {"rootTag":1}
```

- Bottom:** A status bar at the bottom of the screen.

Home.js — ReactNativeFundamentals

iPhone 14
iOS 16.0

Screens > Home.js

```
1 import React from 'react';
2 import {SafeAreaView, StyleSheet, Text, View, Button} from 'react-native';
3
4 export default function Home() {
5   return (
6     <SafeAreaView>
7       <View style={styles.mainContainer}>
8         <Text> Hello Class! </Text>
9         <Button title="Workout" onPress={() => console.log('Change screens')} />
10      </View>
11    </SafeAreaView>
12  );
13}
14 const styles = StyleSheet.create({
15   mainContainer: {
16     display: 'flex',
17     justifyContent: 'center',
18     alignItems: 'center',
19   }
}
```

To open developer menu press "d"

```
BUNDLE ./index.js
LOG | Running "ReactNativeFundamentals" with {"rootTag":1,"initialProps":{}}
BUNDLE ./index.js
LOG | Running "ReactNativeFundamentals" with {"rootTag":1}
LOG | Change screens
```

3:14

Home

Hello Class!

Workout

Ln 8, Col 33 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier ⚡ ⚡

Workout.js — ReactNativeFundamentals

EXPLORER ...

REACTNATIVEFUNDAMENTALS

Screens > Workout.js

Workout.js

1 import React from 'react';
2 import {SafeAreaView, StyleSheet, Text, View, Button} from 'react-native'
3
4 export default function Workout() {
5 return [
6 <SafeAreaView>
7 <View style={styles.mainContainer}>
8 <Text> Workout Screen! </Text>
9 </View>
10 </SafeAreaView>
11];
12 }
13 const styles = StyleSheet.create({
14 mainContainer: {
15 display: 'flex',
16 justifyContent: 'center',
17 alignItems: 'center',
18 height: '100%',
19 width: '100%',
20 backgroundColor: 'yellow',
21 },
22 });
23

_node-version
_ruby-version
.buckconfig
.eslintrc.js
.flowconfig
.gitignore
.prettierrc.js
.watchmanconfig
App.js M
app.json
babel.config.js
Gemfile
index.js
metro.config.js
package-lo... M
package.json M

> OUTLINE
> TIMELINE
> JAVA PROJECTS

master* React Native Packager Ln 5, Col 11 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier

JS App.js

```
1  import React from 'react';
2  import type {Node} from 'react';
3  import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
4  import {NavigationContainer} from '@react-navigation/native';
5  import {createNativeStackNavigator} from '@react-navigation/native-stack';
6  import Home from './Screens/Home';
7  import Workout from './Screens/Workout';
8
9  const Stack = createNativeStackNavigator();
10
11 const App: () => Node = () => {
12   return (
13     <NavigationContainer>
14       <Stack.Navigator initialRouteName="Home">
15         <Stack.Screen name="Home" component={Home} />
16         <Stack.Screen name="Workout" component={Workout} />
17       </Stack.Navigator>
18     </NavigationContainer>
19   );
20 };
21 export default App;
22
```

Screens > **js** Home.js

```
1 import React from 'react';
2 import {SafeAreaView, StyleSheet, Text, View, Button} from 'react-native'
3 import {useNavigation} from '@react-navigation/native';
4
5 export default function Home() {
6   const navigation = useNavigation();
7
8   const handleWorkoutButton = () => {
9     navigation.navigate('Workout');
10  };
11
12  return (
13    <SafeAreaView>
14      <View style={styles.mainContainer}>
15        <Text>Hello Class!</Text>
16        <Button title="Workout" onPress={() => handleWorkoutButton()} />
17      </View>
18    </SafeAreaView>
19  );
20
21  const styles = StyleSheet.create({
22    mainContainer: {
23      display: 'flex',
24      justifyContent: 'center',
25      alignItems: 'center',
26      height: '100%',
27      width: '100%',
28      backgroundColor: 'yellow',
29    },
30  });

```



What do you think is being used under the hood to navigate in Android?

“Intents”

What do you think is being used under the hood to navigate in iOS?

A “Navigation Controller”

AsyncStorage API

(<https://react-native-async-storage.github.io/async-storage/>)



Getting started

Installation

Usage

Manual linking

API

Known limits

Advanced



API

getItem

Gets a string value for given `key`. This function can either return a string value for existing `key` or return `null` otherwise.

In order to store object value, you need to deserialize it, e.g. using `JSON.parse()`.

Note (legacy): you can use optional callback as an alternative for returned promise.

Signature:

```
static getItem(key: string, [callback]: ?(error: ?Error, result: ?string)
```

Returns:

`Promise` resolving with a string value, if entry exists for given `key`, or `null` otherwise.

`Promise` can also be rejected in case of underlying storage error.

`getItem`
`setItem`
`mergeItem`
`removeItem`
`getAllKeys`
`multiGet`
`multiSet`
`multiMerge`
`multiRemove`
`clear`
`useAsyncStorage`



Getting started

Installation

Usage

Manual linking

API

Known limits

Advanced

Debugging

Help

Known storage limits

Android

Android

AsyncStorage for Android uses SQLite for storage backend. While it has its own size limits, Android system also have two known limits: total storage size and per-entry size limit.

- Total storage size is capped at 6 MB by default. You can increase this size by specifying a new size using feature flag.
- Per-entry is limited by a size of a WindowCursor, a buffer used to read data from SQLite. Currently its size is around 2 MB. This means that the single item read at one time cannot be larger than 2 MB. There's no supported workaround from AsyncStorage. We suggest keeping your data lower than that, by chopping it down into many entries, instead of one massive entry. This is where `multiGet` and `multiSet` APIs can shine.

Edit this page

package.json — ReactNativeFundamentals

EXPLORER ... package.json M

REACTNATIVEFUNDAMENTALS

- > @ampproject
- > @babel
- > @bccoe
- > @cnakazawa
- > @eslint
- > @hapi
- > @humanwhocodes
- > @istanbuljs
- > @jest
- > @jriderwell
- > @react-native
- > @react-native-async-storage @react-native-async-storage
- > @react-native-community
- > @react-navigation
- > @sideway
- > @sinclair
- > @sinonjs
- > @toottallnate
- > @types
- > @typescript-eslint
- > abab
 - > short-controller

... package.json > {} dependencies > @react-navigation/native-stack

```
private : true,
> Debug
"scripts": {
  "android": "react-native run-android",
  "ios": "react-native run-ios",
  "start": "react-native start",
  "test": "jest",
  "lint": "eslint ."
},
"dependencies": {
  "@react-native-async-storage/async-storage": "^1.17.10",
  "@react-navigation/native": "^6.0.13",
  "@react-navigation/native-stack": "^6.9.1",
  "react": "18.1.0",
  "react-native": "0.70.4",
  "react-native-safe-area-context": "^4.4.1",
  "react-native-screens": "^3.18.2"
},
"devDependencies": {
  "@babel/core": "7.12.0"
}
```

OUTPUT PROBLEMS TERMINAL ... bash - ios + □ ^ X

● ⚡ → 17:32 ios / npm install @react-native-async-storage/async-storage

```
added 3 packages, and audited 1314 packages in 3s

106 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

○ ⚡ → 17:32 ios /

master* ↻ ⊞ 0 △ 0 ▷ React Native Packager ↻ Ln 15, Col 35 Spaces: 2 UTF-8 LF { JSON ↙ Prettier ↙ ↙

package.json — ReactNativeFundamentals

EXPLORE

REACTNATIVEFUNDAMENTALS

- Package-lock.json
- Screens
 - Home.js
 - Signup.js
 - Workout.js
- _node-version
- _ruby-version
- .buckconfig
- .eslintrc.js
- .flowconfig
- .gitignore
- .prettierrc.js
- .watchmanconfig
- App.js
- app.json
- babel.config.js
- Gemfile
- index.js
- metro.config.js
- package-lock.json
- package.json M

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

```
● ⏱ → 17:44 ReactNativeFundamentals ✘ cd ios
● ⏱ → 17:44 ios ✘ pod install
ignoring RNCAsyncStorage because its extensions are not built. Try: gem pristine f
fi --version 1.15.4
Auto-linking React Native modules for target `ReactNativeFundamentals`: RNCA
syncStorage, RNScreens, and react-native-safe-area-context
[Codegen] Generating ./build/generated/ios/React-Codegen.podspec.json
Analyzing dependencies
[Codegen] Found RNCAutoLinkSpec
Downloading dependencies
Installing RNCAsyncStorage (1.17.10)
Generating Pods project
Setting REACT_NATIVE build settings
Setting CLANG_CXX_LANGUAGE_STANDARD to c++17 on /Users/khalfanibozeman/Desktop
/p/Fall22/Mobile-App-Development/presentation/ReactNativeFundamentals/ios/Reac
tNativeFundamentals.xcodeproj
Pod install took 26 [s] to run
Integrating client project
Pod installation complete! There are 62 dependencies from the Podfile and 52
total pods installed.
● ⏱ → 17:45 ios ✘ cd ..
● ⏱ → 17:45 ReactNativeFundamentals ✘
```

master* ↻ ⊞ 0 △ 0 ▷ React Native Packager ↻ Ln 15, Col 35 Spaces: 2 UTF-8 LF { JSON Prettier 🔍 🔔

App.js — ReactNativeFundamentals

EXPLORER ...

REACTNATIVEFUNDAMENTALS

- > node_modules
- ↳ Screens
 - Home.js
 - Signup.js** U
 - workout.js
- _node-version
- _ruby-version
- .buckconfig
- .eslintrc.js
- .flowconfig
- .gitignore
- .prettierrc.js
- .watchmanconfig
- App.js** M
- {.} app.json
- B babel.config.js
- Gemfile
- index.js
- metro.config.js
- package-lock.json M
- package.json M

OUTLINE TIMELINE JAVA PROJECTS

Signup.js U App.js M X

App.js

```
1 import React from 'react';
2 import type {Node} from 'react';
3 import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
4 import {NavigationContainer} from '@react-navigation/native';
5 import {createNativeStackNavigator} from '@react-navigation/native';
6 import Home from './Screens/Home';
7 import Workout from './Screens/Workout';
8 import Signup from './Screens/Signup'
9
10 const Stack = createNativeStackNavigator();
11
12 const App: () => Node = () => {
13   return (
14     <NavigationContainer>
15       <Stack.Navigator initialRouteName="Signup">
16         <Stack.Screen name="Home" component={Home} />
17         <Stack.Screen name="Workout" component={Workout} />
18         <Stack.Screen name="Signup" component={Signup} />
19       </Stack.Navigator>
20     </NavigationContainer>
21   );
22 }
23
24 export default App;
```

Ln 11, Col 1 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier

Screens > JS Signup.js

```
13
14  export default function Signup() {
15    return (
16      <SafeAreaView>
17        <TouchableWithoutFeedback onPress={Keyboard.dismiss}>
18          <View style={styles.mainContainer}>
19            <TextInput style={styles.textInput} placeholder="Name" />
20            <TextInput style={styles.textInput} placeholder="Email" />
21            <Button title="register" />
22          </View>
23        </TouchableWithoutFeedback>
24      </SafeAreaView>
25    );
26  }
27  const styles = StyleSheet.create({
28    mainContainer: {
29      display: 'flex',
30      justifyContent: 'center',
31      alignItems: 'center',
32      height: '100%',
33      width: '100%'.
34    },
35    textInput: {
36      height: 50,
37      width: 200,
38      borderColor: 'black',
39      borderWidth: 1,
40      borderRadius: 3,
41      padding: 3,
42      margin: 5,
```

5:58



Signup

Name

Email

[register](#)

```
12 import React, {useState, useEffect} from 'react';
13 export default function Signup() {
14     const [name, setName] = useState('');
15     const [email, setEmail] = useState('');
16
17     const nameHandler = text => {
18         setName(text);
19     };
20     const emailHandler = text => {
21         setEmail(text);
22     };
23
24     useEffect(() => {
25         console.log('RENDER:');
26         console.log(name);
27         console.log(email);
28     }, []);
29
30     return (
31         <SafeAreaView>
32             <TouchableWithoutFeedback onPress={Keyboard.dismiss}>
33                 <View style={styles.mainContainer}>
34                     <TextInput
35                         onChangeText={text => {
36                             nameHandler(text);
37                         }
38                         style={styles.textInput}
39                         placeholder="Name"
40                         />
41                     <TextInput
42                         onChangeText={text => {
43                             emailHandler(text);
44                         }
45                         style={styles.textInput}
46                         placeholder="Email"
47                         />
48                 </View>
49             </SafeAreaView>
50         );
51     );
52 }
```

Signup.js — ReactNativeFundamentals

iPhone 14
iOS 16.0

Signup

6:10

Name

Email

register

Signup.js

Screens > Signup.js

```
12 import React, {useState, useEffect} from 'react';
13 export default function Signup() {
14   const [name, setName] = useState();
15   const [email, setEmail] = useState();
16
17   const nameHandler = text => {
18     setName(text);
19   };
20   const emailHandler = text => {
21     setEmail(text);
22   };
23   useEffect(() => {
24     console.log('RENDER:');
25     console.log(name);
26     console.log(email);
27   }, []);
28   return (
29     <SafeAreaView>
```

BUNDLE ./index.js

```
LOG Running "ReactNativeFundamentals" with {"rootTag":51,"initialProps":{}}
LOG RENDER:
LOG undefined
LOG undefined
```

39 <TextInput
40 onChangeText={text => {
41 emailHandler(text);

master* 0 △ 0 React Native Packager Ln 22, Col 5 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂

Screens > Signup.js

```
12 import React, {useState, useEffect} from 'react'
13 export default function Signup() {
14   const [name, setName] = useState();
15   const [email, setEmail] = useState();
16
17   const nameHandler = text => {
18     setName(text);
19   };
20   const emailHandler = text => {
21     setEmail(text);
22   };
23   useEffect(() => {
24     console.log('RENDER:');
25     console.log(name);
26     console.log(email);
27   }, []);
28   return (
29     <SafeAreaView>
30       <Text>Khal</Text>
31     </SafeAreaView>
32   );
33 }
```

khalfanibozeman — Metro

```
LOG Khaltttt
LOG undefined
info Reloading app...
BUNDLE ./index.js
```

LOG Running "ReactNativeFundamentals" with {"rootTag":61,'

LOG RENDER:

LOG undefined

LOG undefined

iPhone 14
iOS 16.0

6:13

Signup

Khal

"Khal"

q w e r t y u i o p

a s d f g h j k l

z x c v b n m ↻

123 space return

39 <TextInput
40 onChangeText={text => {
41 emailHandler(text);

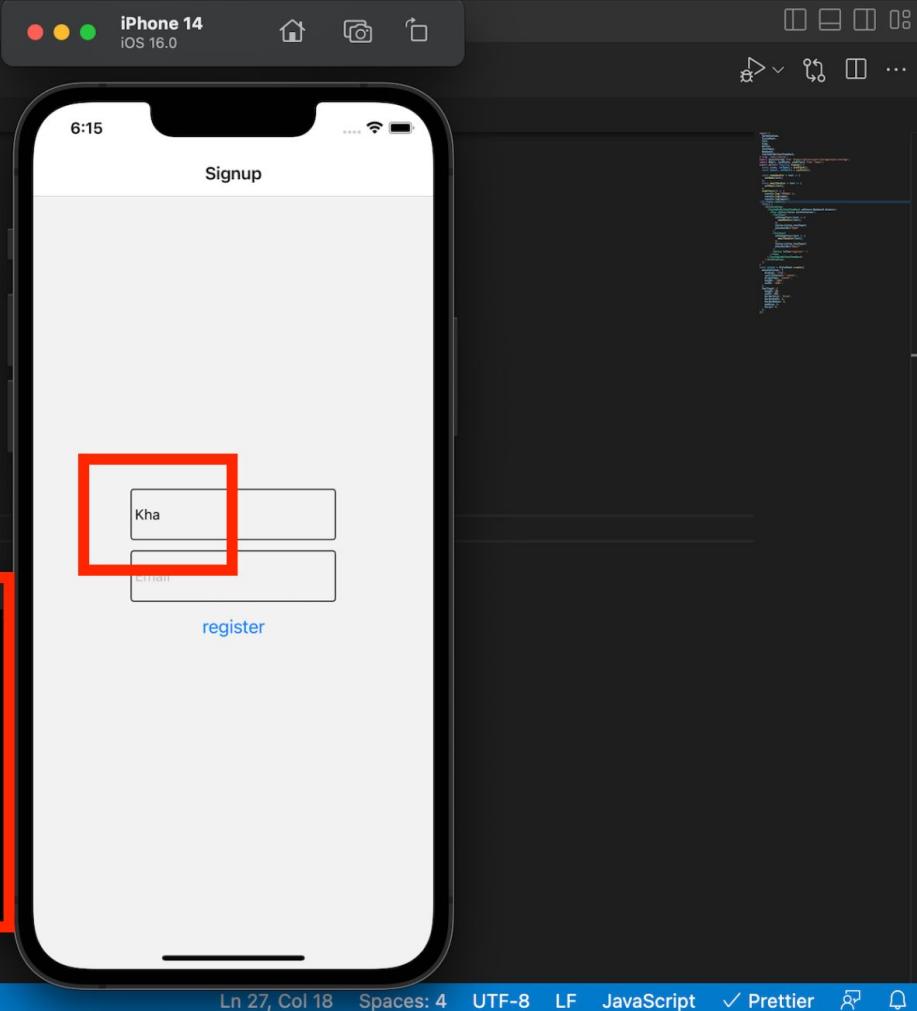
master* ↻ ⌂ ⌂ 0 △ 0 ▷ React Native Packager ↻ Ln 27, Col 7 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂

JS Signup.js X

```
Screens > JS Signup.js
14 const [name, setName] = useState();
15 const [email, setEmail] = useState();
16
17 const nameHandler = text => {
18   setName(text);
19 };
20 const emailHandler = text => {
21   setEmail(text);
22 };
23 useEffect(() => {
24   console.log('RENDER:');
25   console.log(name);
26   console.log(email);
27 }, [name, email]);
28 return (

```

```
khalfanibozeman — Metro
LOG Running "ReactNativeFundamentals" with {"rootTag":"
LOG RENDER:
LOG undefined
LOG undefined
LOG RENDER:
LOG K
LOG undefined
LOG RENDER:
LOG Kh
LOG undefined
LOG RENDER:
LOG Kha
LOG undefined
```



```
const emailHandler = text => {
  setEmail(text);
};

// Save the data locally:
const saveLocalData = async () => {
  try {
    await AsyncStorage.setItem('name', name);
    await AsyncStorage.setItem('email', email);
  } catch (error) {
    console.log(error);
  }
  navigation.navigate('Home');
};
```

```
  placeholder="Email"
/>
<Button
  onPress={() => {
    saveLocalData();
  }}
  title="register"
/>
```

```
};

// Fetch local data:
const getLocalData = async () => {
  try {
    const name = await AsyncStorage.getItem(`name`);
    const email = await AsyncStorage.getItem(`email`);
    console.log(name);
    console.log(email);
    if (email != null && name != null) navigation.navigate('Home');
  } catch (error) {
    console.log(error);
  }
};

useEffect(() => {
  if (isLoading) {
    getLocalData();
    setIsLoading(false);
  }
}, []);

return (

```

```
LOG | Khalfani
LOG | Khalfani@gsu.edu
info Reloading app...
BUNDLE ./index.js

LOG | Running "ReactNativeFundamentals" with {"rootTag":181,"initialProps":{}}
LOG | Khalfani
LOG | Khalfani@gsu.edu
info Reloading app...
BUNDLE ./index.js

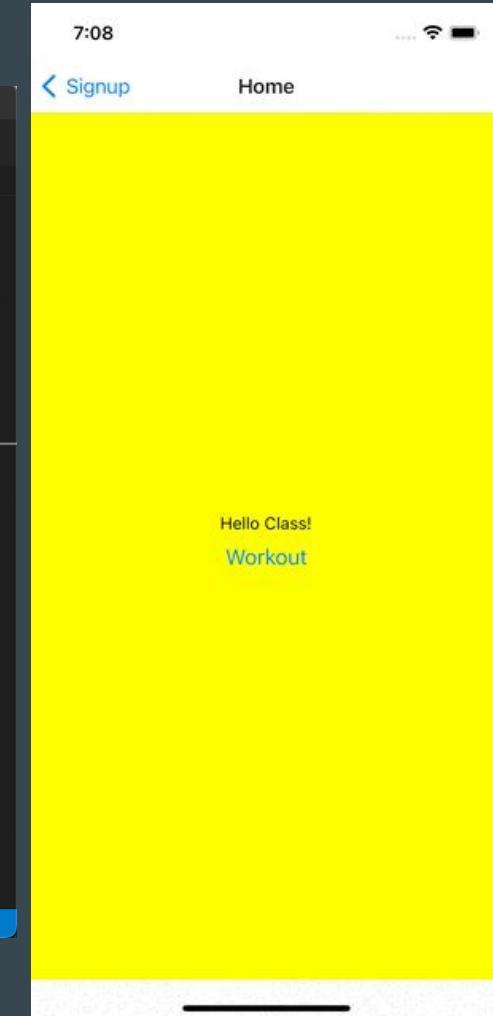
LOG | Running "ReactNativeFundamentals" with {"rootTag":191,"initialProps":{}}
LOG | Khalfani
LOG | Khalfani@gsu.edu
```

Home.js
Signup.js U
Workout.js
_node-version
_ruby-version
.buckconfig
.eslintrc.js
.flowconfig
.gitignore
.prettierrc.js
.watchmanconfig
App.js M
{} app.json
babel.config.js
> OUTLINE
> TIMELINE
> JAVA PROJECTS

Signup.js – ReactNativeFundamentals

```
Screens > Signup.js
32     } catch (error) {
33     |   console.log(error);
34     }
35     navigation.navigate('Home');
36   };
37   // Fetch local data:
38   const getLocalData = async () => {
39     try {
40       const name = await AsyncStorage.getItem(`name`);
41       const email = await AsyncStorage.getItem(`email`);
42       console.log(name);
43       console.log(email);
44       if (email != null && name != null) navigation.navigate('Home');
45     } catch (error) {
46       |   console.log(error);
47     }
48   };
49
50   useEffect(() => {
51     if (isLoading) {
52       getLocalData();
53       setIsLoading(false);
54     }
55   }, [()]);
56   return (
57     <SafeAreaView>
58       <TouchableWithoutFeedback onPress={Keyboard.dismiss}>
59         <View style={styles.mainContainer}>
60           <Text style={styles.title}>Hello Class!</Text>
61           <Text style={styles.title}>Workout</Text>
62         </View>
63       </TouchableWithoutFeedback>
64     </SafeAreaView>
65   );
66 }

Ln 47, Col 6 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂
```



Geolocation API

(<https://github.com/Agontuk/react-native-geolocation-service>)



Geolocation

Deprecated. Use one of the community packages instead.

The Geolocation API extends the [Geolocation web spec](#).

As a browser polyfill, this API is available through the `navigator.geolocation` global - you do not need to `import` it.

On Android, this uses the `android.location` API. This API is not recommended by Google because it is less accurate and slower than the recommended [Google Location Services API](#). In order to use it with React Native, use the `react-native-geolocation-service` module.



APIs

AccessibilityInfo

Alert

Animated.Value

Animated.ValueXY

Appearance

AppRegistry

AppState

DevSettings

Dimensions

Easing

InteractionManager

Keyboard

LayoutAnimation

Linking

PanResponder

PixelRatio

Platform

PlatformColor



Platform

Example

Platform API Example

```
import React from 'react';
import { Platform, StyleSheet, Text, ScrollView } from
  'react-native';

const App = () => {
  return (
    <ScrollView contentContainerStyle={styles.container}>
      <Text>OS</Text>
      <Text style={styles.value}>{Platform.OS}</Text>
      <Text>OS Version</Text>
      <Text style={styles.value}>{Platform.Version}</Text>
      <Text>isTV</Text>
      <Text style={styles.value}>{Platform.isTV.toString()}</Text>
      <Text>
        {Platform.OS === 'ios' && <br>
          <Text>isPad</Text>
          <Text style={styles.value}>
            {JSON.stringify(Platform.constants, null, 2)}</Text>
        </Text>
      </Text>
      <Text>Constants</Text>
      <Text style={styles.value}>
        {JSON.stringify(Platform.constants, null, 2)}</Text>
      </Text>
    </ScrollView>
  );
}

export default App;
```



Download [Expo Go](#) and scan
the QR code to get started.



Connected devices 0



[Log in](#) or set a Device ID to open this Snack

Example

Properties

constants

isPad •

isTV

isTesting

OS

Version

Methods

select()



Keyboard

LayoutAnimation

Linking

PanResponder

PixelRatio

Platform

PlatformColor

NEW

RootTag

Share

StyleSheet

Systrace

Transforms

Vibration

Hooks



useColorScheme

useWindowDimensions

Android APIs



BackHandler

PermissionsAndroid

ToastAndroid

PermissionsAndroid

Project with Native Code Required

The following section only applies to projects with native code exposed. If you are using the managed Expo workflow, see the guide on [Permissions](#) in the Expo documentation for the appropriate alternative.

PermissionsAndroid provides access to Android M's new permissions model. The so-called "normal" permissions are granted by default when the application is installed as long as they appear in `AndroidManifest.xml`. However, "dangerous" permissions require a dialog prompt. You should use this module for those permissions.

On devices before SDK version 23, the permissions are automatically granted if they appear in the manifest, so `check` should always result to `true` and `request` should always resolve to `PermissionsAndroid.RESULTS.GRANTED`.

If a user has previously turned off a permission that you prompt for, the OS will advise your app

Example

Permissions that require prompting the user

Result strings for requesting permissions

Methods

`constructor()`

`check()`

`request()`

`requestMultiple()`

```
import React, {useState, useEffect} from 'react';
import {
  SafeAreaView,
  StyleSheet,
  Text,
  View,
  Button,
  Platform,
  PermissionsAndroid,
} from 'react-native';
import {useNavigation} from '@react-navigation/native';
import Geolocation from 'react-native-geolocation-service';

export default function Home() {
  // Local navigation reference:
  const navigation = useNavigation();
  const [isLoading, setIsLoading] = useState(true);
  const [location, setLocation] = useState(null);
```

```
const requestIphoneLocation = async () => {
  const access = await Geolocation.requestAuthorization('always').then(
    res => {
      if (res == 'granted') {
        console.log('IOS location enabled');
        Geolocation.getCurrentPosition(
          position => {
            console.log(position);
            setLocation(position);
          },
          error => {
            console.log(error.code, error.message);
          },
          {enableHighAccuracy: true, timeout: 15000, maximumAge: 10000},
        );
      }
    },
  );
};
```

Info.plist — ReactNativeFundamentals

EXPLORER ... Signup.js U Home.js M Info.plist M X

ios > ReactNativeFundamentals > Info.plist

```
<dict>
    <key>localhost</key>
    <dict>
        <key>NSExceptionAllowsInsecureHTTPLoads</key>
        <true/>
    </dict>
</dict>
</dict>
<key>NSLocationAlwaysAndWhenInUseUsageDescription</key>
<string>Please allow us to use your location to calculate</string>
<key>NSLocationAlwaysUsageDescription</key>
<string>Please allow us to use your location to calculate</string>
<key>NSLocationWhenInUseUsageDescription</key>
<string>Please allow us to use your location to calculate</string>
<key>UILaunchStoryboardName</key>
<string>LaunchScreen</string>
<key>UIRequiredDeviceCapabilities</key>
<string>armv7</string>
```

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

khalfanibozeman — Metro

To reload the app press "r"
To open developer menu press "a"

BUNDLE ./index.js

LOG Running "ReactNativeFundamentals" with {"rootTag":1,"initialProps":{}}

LOG Khalfani

LOG Khalfani@gsu.edu

iPhone 14 iOS 16.0

7:58 Home

Allow "ReactNativeFundamentals" to use your location?
Allowing the application to track your location is how distance, pace and other metrics are able to be calculated.

Precise: On

Allow Once
Allow While Using App
Don't Allow

Ln 43, Col 92 Tab Size: 4 UTF-8 LF XML ⚡ Prettier

```
const requestAndroidLocation = async () => {
  try {
    let access = await PermissionsAndroid.request(
      PermissionsAndroid.PERMISSIONS.ACCESS_FINE_LOCATION,
      {
        title: 'Geolocation Permission',
        message: 'Can we access your location?',
        buttonNeutral: 'Ask Me Later',
        buttonNegative: 'Cancel',
        buttonPositive: 'OK',
      },
    );
    if (access == 'granted') {
      console.log('Android location enabled');
      Geolocation.getCurrentPosition([
        position => {
          console.log(position);
          setLocation(position);
        },
        error => {
          console.log(error.code, error.message);
        },
        {enableHighAccuracy: true, timeout: 15000, maximumAge: 10000},
      ]);
    }
  } catch (err) {
    console.log(err);
  }
};
```

The screenshot shows the Android Studio interface with two main windows. On the left, the Emulator window displays a location permission dialog for the app "ReactNativeFundamentals". The dialog asks if the app can access the device's location, with options: "While using the app", "Only this time", and "Deny". A red box highlights this dialog. On the right, the code editor window shows the "AndroidManifest.xml" file. A red box highlights the permissions section, which includes four `<uses-permission` tags for INTERNET, ACCESS_BACKGROUND_LOCATION, ACCESS_COARSE_LOCATION, and ACCESS_FINE_LOCATION. The full XML code is as follows:

```
1 manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.reactnat
2 
3 <uses-permission android:name="android.permission.INTERNET" />
4 <uses-permission android:name="android.permission.ACCESS_BACKGROUND_LOCATION" />
5 <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
6 <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
7 
8 <activity android:name=".MainActivity" android:label="@string/app_name" android:config
9   <intent-filter>
10     <action android:name="android.intent.action.MAIN" />
11     <category android:name="android.intent.category.LAUNCHER" />
12   </intent-filter>
13 </activity>
14 </application>
15 </manifest>
16 
```

Home.js — ReactNativeFundamentals

Emulator: Pixel API Push... ↻ K

Device Manager

```
useEffect(() => {
  if (isLoading) {
    if (Platform.OS == 'ios') requestiPhoneLocation();
    if (Platform.OS == 'android') requestAndroidLocation();
    setIsLoading(false);
  }
});
```

12:31

Signup Home

Home

Workout

khalfanibozeman — Metro

```
LOG Running "ReactNativeFundamentals" with {"rootTag":411,"initialProps":{}}
LOG Khalfani
LOG Khalfani@gsu.edu
LOG IOS location enabled
LOG {"coords": {"accuracy": 10, "altitude": 0, "altitudeAccuracy": -1, "heading": 277.39, "latitude": 37.3306482, "longitude": -122.02998145, "speed": 3.61}, "timestamp": 1667842266506.6772}
LOG Running "ReactNativeFundamentals" with {"rootTag":11}
LOG Rick James
LOG rick@gsu.edu
LOG Android location enabled
LOG {"coords": {"accuracy": 5, "altitude": 0, "altitudeAccuracy": 0.5, "heading": 0, "latitude": 37.8199283, "longitude": -122.478255, "speed": 0}, "mocked": false, "provider": "fused", "timestamp": 1667842140180}
```

Event Log Layout Inspector

2 master ackager ↻ Ln 45, Col 13 Spaces: 2 UTF-8 LF JavaScript Prettier ↻

```
export default function Home() {
  // Local navigation reference:
  const navigation = useNavigation();
  const [isLoading, setIsLoading] = useState(true);
  const [location, setLocation] = useState(null);

  // Button press handler:
  const handleWorkoutButton = () => {
    navigation.navigate('Workout', {
      location: location,
    });
  };

  const requestAndroidLocation = async () => {
    try {
```

```
export default function Workout({route, navigation}) {
  const [location] = useState(route.params.location);

  useEffect(() => {
    console.log(location);
  });

  return (
    <SafeAreaView>
      <View style={styles.mainContainer}>
        <Text> Workout Screen! </Text>
```

Workout.js — ReactNativeFundamentals

iPhone 14
iOS 16.0

Emulator: Pixel API 30

Workout

```
> Js Workout.js
  Button,
} from 'react-native';

export default function Workout({route, navigation}) {
  const [location] = useState(route.params.location);

  useEffect(() => {
    console.log(Platform.OS + ': ');
    console.log(location);
  });
}
```

khalfanibozeman — Metro

```
g": 0, "latitude": 37.8199283, "longitude": -122.478255, "speed": 0}, "mocked": false, "provider": "fused", "timestamp": 1667842693545}
-
-
-
-
-
LOG ios:
LOG {"coords": {"accuracy": 5, "altitude": 0, "altitudeAccuracy": -1, "heading": 281.73, "latitude": 37.3246192, "longitude": -122.01989178, "speed": 3.86}, "timestamp": 1667842759831.0242}
LOG android:
LOG {"coords": {"accuracy": 5, "altitude": 0, "altitudeAccuracy": 0.5, "heading": 0, "latitude": 37.8199283, "longitude": -122.478255, "speed": 0}, "mocked": false, "provider": "fused", "timestamp": 1667842693545}
```

Workout Screen!

Workout Screen!

Event Log Layout Inspector

Packager

Ln 16, Col 25 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier

What is the (native) object use to gather location information on Android?

“LocationManager”

What two frameworks does iOS provide for accessing location information?

“CoreLocation” & “MapKit”

Maps

(<https://github.com/react-native-maps/react-native-maps>)

Workout.js — ReactNativeFundamentals

EXPLORER ... Home.js Workout.js M X

Screens > js Workout.js

```
8   |   Button,
9   | } from 'react-native';
10  | import Geolocation from 'react-native-geolocation-service';
11  | import MapView from 'react-native-maps';
12  |
```

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

bash - ios + ▾

→ 13:17 ReactNativeFundamentals ✘ npm install react-native-maps

```
added 2 packages, and audited 1317 packages in 14s

106 packages are looking for funding
  run `npm fund` for details
```

Found 0 vulnerabilities

→ 13:17 ReactNativeFundamentals ✘ cd ios

→ 13:18 ios ✘ pod install

```
Ignoring ffi-1.15.4 because its extensions are not built. Try: gem pristine ffi --version 1.15.4
Auto-linking React Native modules for target `ReactNativeFundamentals`: RNCAsyncStorage, RNStorage,
react-native-geolocation-service, react-native-maps, and react-native-safe-area-context
[Codegen] Generating ./build/generated/ios/React-Codegen.podspec.json
Analyzing dependencies
[Codegen] Found FBReactNativeSpec
Downloading dependencies
Installing react-native-maps (1.3.2)
Generating Pods project
Setting REACT_NATIVE build settings
Setting CLANG_CXX_LANGUAGE_STANDARD to c++17 on /Users/khalfanibozeman/Desktop/Fall22/Mobile-App-Development/presentation/ReactNativeFundamentals/ios/ReactNativeFundamentals.xcodeproj
Pod install took 27 [s] to run
Integrating client project
Pod installation complete! There are 64 dependencies from the Podfile and 54 total pods installed.
```

→ 13:18 ios ✘

master* 0 0 ▾ 0 D React Native Packager

Ln 12, Col 1 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier

4

Android

Specify your Google Maps API key

Add your API key to your manifest file (`android/app/src/main/AndroidManifest.xml`):

```
<application>
    <!-- You will only need to add this meta-data tag, but make sure it's a child of application -->
    <meta-data
        android:name="com.google.android.geo.API_KEY"
        android:value="Your Google maps API Key Here"/>
</application>
```

(At this point, we are going to focus on iPhone. No API key needed!)

Map.js — ReactNativeFundamentals

EXPLORER ...

REACTNATIVEFUNDAMENTALS

- > Pods
- > ReactNativeFundamentals
- > ReactNativeFundamenta...
- > ReactNativeFundamenta...
- > ReactNativeFundamenta...
- > _xcode.env
- Podfile
- Podfile.lock M
- > node_modules
- > Screens
- js Home.js
- js Map.js U
- js Signup.js
- js Workout.js M

_node-version

_ruby-version

.buckconfig

.eslintrc.js

.flowconfig

.gitignore

.prettierrc.js

OUTLINE

TIMELINE

JAVA PROJECTS

Screens > js Map.js

```
1 import React from 'react';
2 import {StyleSheet, View} from 'react-native';
3 import MapView from 'react-native-maps';
4 // Props accept data from one component to another
5 export default function Map(props) {
6   return (
7     <View style={styles.container}>
8       <MapView
9         style={styles.map}
10        // Could be always true in production
11        followUserLocation={props.followUserLocation}
12        showsUserLocation={props.showUserLocation}
13        // Initial map loads to the device location
14        initialRegion={{
15          latitude: props.location.coords.latitude,
16          longitude: props.location.coords.longitude,
17          // Zoom in or out:
18          latitudeDelta: 0.0222,
19          longitudeDelta: 0.0121,
20        }}
21      />
22    </View>
23  );
24}
// * Improper styling = invisible map!
25 const styles = StyleSheet.create({
26   container: {
27     ...StyleSheet.absoluteFillObject,
28     height: 400,
29     width: 400,
```

Ln 2, Col 47 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂

```
import Geolocation from 'react-native';
import Map from './Map';
```

```
return [
  <SafeAreaView>
    <View style={styles.mainContainer}>
      <View style={styles.mapContainer}>
        <Map
          location={location}
          showUserLocation={showUserLocation}
          followUserLocation={followUserLocation}
          markers={markers}
        />
      </View>
```

```
const [totalDistance, setTotalDistance] = useState(0);
const [showUserLocation, setShowUserLocation] = useState(false);
const [followUserLocation, setFollowUserLocation] = useState(false);

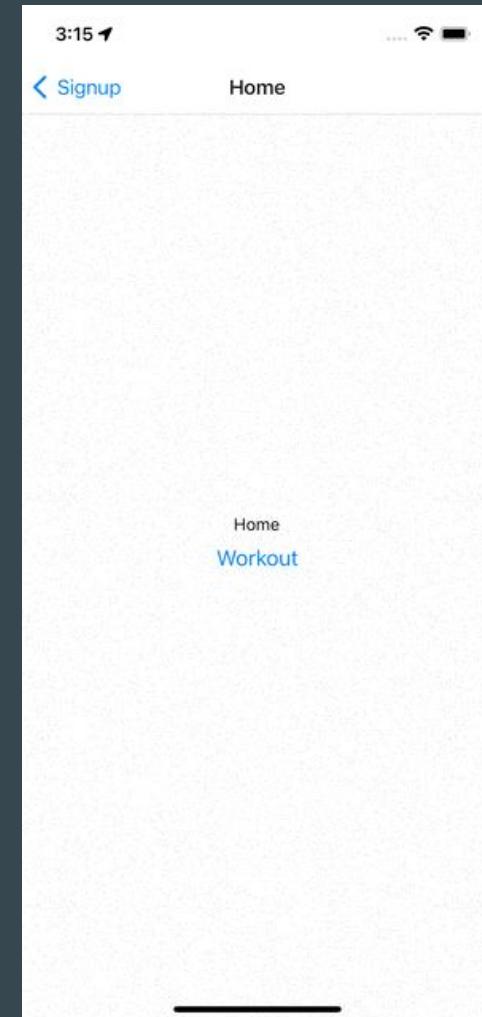
const startRun = () => {
    calculateDistance();
    setFollowUserLocation(true);
    setShowUserLocation(true);
};
```

```
const calculateDistance = () => {
  Geolocation.getCurrentPosition(
    position => {
      //https://www.geeksforgeeks.org/program-distance-two-points-earth/
      lon1 = (position.coords.longitude * Math.PI) / 180;
      lon2 = (position.coords.longitude * Math.PI) / 180;
      lat1 = (position.coords.latitude * Math.PI) / 180;
      lat2 = (position.coords.latitude * Math.PI) / 180;

      // Haversine formula:
      let dlon = lon2 - lon1;
      let dlat = lat2 - lat1;
      let a =
        Math.pow(Math.sin(dlat / 2), 2) +
        Math.cos(lat1) * Math.cos(lat2) * Math.pow(Math.sin(dlon / 2), 2);
      let c = 2 * Math.asin(Math.sqrt(a));
      // Radius of earth in miles:

      let r = 3956;

      // calculate the result + distance already traveled
      distance = c * r + totalDistance;
      console.log(distance);
      setTotalDistance(distance);
      // update current location
      setLocation(position);
    },
    error => {
      console.log(error.code, error.message);
    },
    {enableHighAccuracy: true, timeout: 15000, maximumAge: 10000},
  );
};
```



Map.js — ReactNativeFundamentals

EXPLORER ... Map.js U X Workout.js M

Screens > Map.js

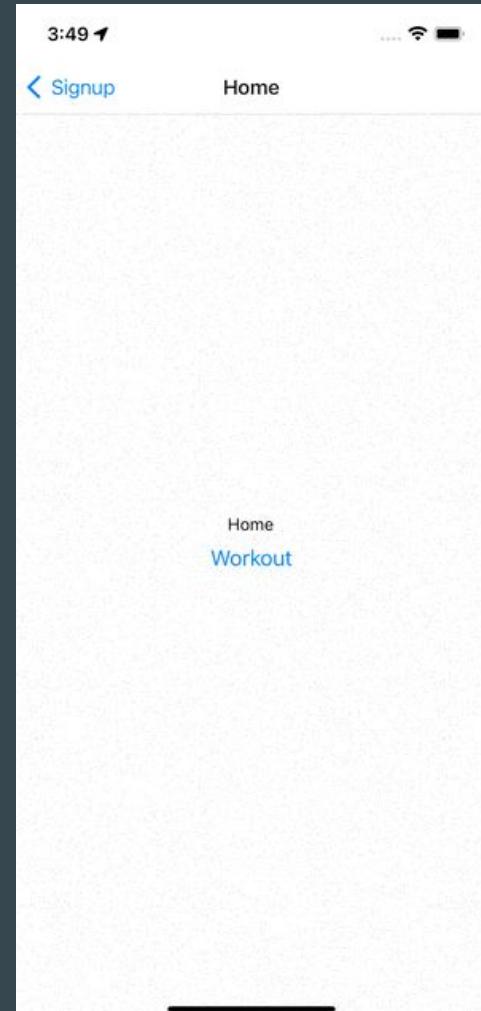
```
<MapView
  style={styles.map}
  // Could be always true in production
  followUserLocation={props.followUserLocation}
  showsUserLocation={props.showUserLocation}
  // Initial map loads to the device location
  initialRegion={{
    latitude: props.location.coords.latitude,
    longitude: props.location.coords.longitude,
    // Zoom in or out:
    latitudeDelta: 0.0052,
    longitudeDelta: 0.0021
  }}>
  <Polyline
    coordinates={props.markers}
    strokeColor="#000" // fallback for when `strokeColors` is not supported by the browser
    strokeColors={[
      '#7F0000',
      '#7F0000', // no color, creates a "long" gradient between the previous and next colors
      '#7F0000',
      '#00D084',
    ]}
    strokeWidth={4}
  />
</MapView>
</View>
);
```

// * Improper styling = invisible map!

const styles = StyleSheet.create({

Ln 28, Col 38 Spaces: 4 UTF-8 LF JavaScript ✓ Prettier

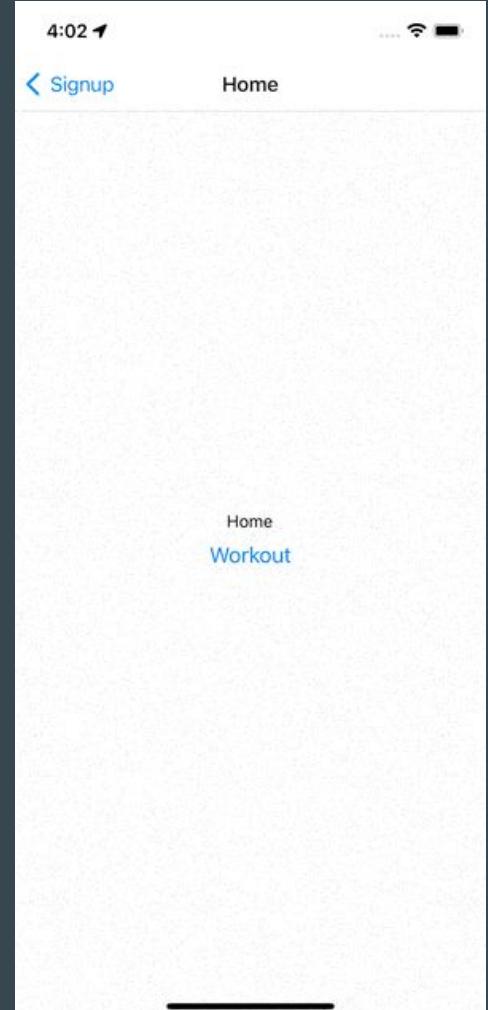
```
// calculate the result + distance already traveled
distance = c * r + totalDistance;
console.log(distance);
setTotalDistance(distance);
// update current location
setLocation(position);
const latLng = {
  latitude: position.coords.latitude,
  longitude: position.coords.longitude,
};
// Update array of markers:
setMarkers([...markers, latLng]);
},
error => {
  console.log(error.code, error.message);
},
{enableHighAccuracy: true, timeout: 15000, maximumAge: 10000},
);
```



```
const startRun = () => {
  calculateDistance();
  setFollowUserLocation(true);
  setShowUserLocation(true);

  setInterval(() => {
    calculateDistance();
  }, 1000);
};

};
```



What about when the screen locks?

Background Timer

(<https://www.npmjs.com/package/react-native-background-timer>)

package.json — ReactNativeFundamentals

EXPLORER ... Map.js Workout.js Home.js M package.json M X

REACTNATIVEFUNDAMENTALS

Screens

- Home.js
- M
- Map.js
- Signup.js
- Workout.js
- _node-version
- _ruby-version
- .buckconfig
- .eslintrc.js
- .flowconfig
- .gitignore
- .prettierrc.js
- .watchmanconfig
- App.js
- app.json
- B babel.config.js
- G Gemfile
- JS index.js
- JS metro.config.js
- JS package-lock.json M
- JS package.json M

package.json > {} dependencies > @react-navigation/native-stack

```
11 "dependencies": {  
12   "@react-native-async-storage/async-storage": "^1.17.10",  
13   "@react-navigation/native": "^6.0.13",  
14   "@react-navigation/native-stack": "^6.9.1",  
15   "react": "18.1.0",  
16   "react-native": "0.70.4",  
17   "react-native-background-timer": "^2.4.1",  
18   "react-native-geolocation-service": "^5.3.1",  
19   "react-native-maps": "1.3.2",  
20   "react-native-safe-area-context": "4.4.1",  
21   "react-native-screens": "3.18.2"  
22 }  
23  
24 "devDependencies": {  
25   "@babel/core": "^7.12.9",  
26   "@babel/runtime": "^7.12.5",  
27   "@react-native-community/eslint-config": "2.0.0",  
28   "babel-jest": "26.6.3",  
29   "eslint": "7.32.0"  
30 }
```

OUTPUT PROBLEMS TERMINAL DEBUG CONSOLE

ruby - ios

```
● ⚡ -> 10:56 ReactNativeFundamentals ✘ npm install react-native-background-timer --save  
added 1 package, and audited 1318 packages in 7s  
106 packages are looking for funding  
  run `npm fund` for details  
found 0 vulnerabilities  
● ⚡ -> 10:57 ReactNativeFundamentals ✘ cd ios  
○ ⚡ -> 10:57 ios ✘ pod install  
ignoring ffi 11.1.1 (because its extensions are not built). Try: gem pristine ffi --version 11.1.1
```

master* ↻ ⊗ 0 △ 0 ▷ React Native Packager ↻

Ln 15, Col 35 Spaces: 2 UTF-8 LF JSON Prettier

Info.plist — ReactNativeFundamentals

EXPLORER ... Info.plist M X

REACTNATIVEFUNDAMENTALS ios > ReactNativeFundamentals > Info.plist

settings.gradle 48
ios 49
 > build
 > Pods
 ReactNativeFu... 50
 > Images.xcassets
 h AppDelegate.h
 C AppDelegate.mm
Info.plist M 51
LaunchScreen.storyboard 52
main.m 53
ReactNativeFundam... 54
ReactNativeFundam... 55
ReactNativeFundam... 56
 !t _xcode.env
Podfile 57
Podfile.lock M 58
node_modules 59
Screens 60
 Home.js M
 Map.js
OUTLINE
TIMELINE
XML DOCUMENT
JAVA PROJECTS

```
<string>armv7</string>
</array>
<key>UISupportedInterfaceOrientations</key>
<array>
<string>UIInterfaceOrientationPortrait</string>
<string>UIInterfaceOrientationLandscapeLeft</string>
<string>UIInterfaceOrientationLandscapeRight</string>
</array>
<key>UIViewControllerBasedStatusBarAppearance</key>
<false/>
<key>UIBackgroundModes</key>
<array>
<string>fetch</string>
<string>audio</string>
<string>location</string>
</array>
</dict>
</plist>
```

master* ⏪ ⏴ 0 △ 0 ▷ React Native Packager ⏪ Ln 45, Col 38 Tab Size: 4 UTF-8 LF XML ⓧ Prettier ⏪ ⏪

```
const startRun = () => {
  setFollowUserLocation(true);
  setShowUserLocation(true);
  setRunStarted(true);
  // Watch the runner
  Geolocation.watchPosition(
    position => {
      // Calculate distance:
      // https://www.geeksforgeeks.org/program-distance-two-points-earth/
      lon1 = (position.coords.longitude * Math.PI) / 180;
      lon2 = (position.coords.longitude * Math.PI) / 180;
      lat1 = (position.coords.latitude * Math.PI) / 180;
      lat2 = (position.coords.latitude * Math.PI) / 180;
      // Haversine formula:
      let dlon = lon2 - lon1;
      let dlat = lat2 - lat1;
      let a =
        Math.pow(Math.sin(dlat / 2), 2) +
        Math.cos(lat1) * Math.cos(lat2) * Math.pow(Math.sin(dlon / 2), 2);
      let c = 2 * Math.asin(Math.sqrt(a));
      // Radius of earth in miles:
      let r = 3956;
      // distance + distance already traveled
      distance = c * r + totalDistance;

      setTotalDistance(distance);
      setLocation(position);
    },
    error => {
      console.log(error.code, error.message);
    },
    {
      enableHighAccuracy: true,
      distanceFilter: 10, // Meters
    },
  );
};
```

```
let time = 0;
BackgroundTimer.runBackgroundTimer(() => {
  time += 1;
  setGlobalTimer(time);
}, 1000);
};

const stopRun = () => {
  setShowUserLocation(false);
  setFollowUserLocation(false);
  setRunStarted(false);
  Geolocation.stopObserving();
  BackgroundTimer.stopBackgroundTimer();
};
```

```
useEffect(() => {
  console.log('[RENDERING]: Workout Screen (' + Platform.OS + ')');
  if (runStarted) {
    const latLng = {
      latitude: location.coords.latitude,
      longitude: location.coords.longitude,
    };
    setMarkers([...markers, latLng]);
  }

  console.log('Distance: ' + totalDistance.toFixed(3) + ' miles');
  console.log('Total time (seconds): ' + globalTimer);

  return () => {
    console.log('Unmounting: ');
    Geolocation.clearWatch();
  };
}, [location]);
```

9:38



Signup

Home

Home

Workout

Diet

What is the first method executed when a user stops directly interacting with an Android app?

“onPause”

What method is executed when a running iOS app is no longer visible?

“applicationDidEnterBackground”

Fetch API

(<https://reactnative.dev/docs/network>)



- The Basics >
- Environment setup >
- Workflow >
 - Running On Device
 - Fast Refresh
 - Debugging
 - Symbolicating a stack trace
 - Testing
 - Using Libraries
 - Using TypeScript
 - Upgrading to new versions
- Connectivity >
 - Networking
 - Security
- Design >
- Performance >
- JavaScript Runtime >
- Native Modules >
- Native Components >
- The New Architecture >

Networking

Many mobile apps need to load resources from a remote URL. You may want to make a POST request to a REST API, or you may need to fetch a chunk of static content from another server.

Using Fetch

React Native provides the Fetch API for your networking needs. Fetch will seem familiar if you have used XMLHttpRequest or other networking APIs before. You may refer to MDN's guide on Using Fetch for additional information.

Making requests

In order to fetch content from an arbitrary URL, you can pass the URL to fetch:

```
fetch('https://mywebsite.com/mydata.json');
```

Fetch also takes an optional second argument that allows you to customize the HTTP request. You may want to specify additional headers, or make a POST request:

- Using Fetch
- Making requests
- Handling the response
- Using Other Networking Libraries
- WebSocket Support
- Known Issues with fetch and cookie based authentication
- Configuring NSURLConnection on iOS

App.js — ReactNativeFundamentals

EXPLORER ... Diet.js U App.js M X

REACTNATIVEFUNDAMENTALS

- _xcode.env
- Podfile
- Podfile.lock M
- > node_modules
- Screens

 - Diet.js U
 - Home.js M
 - Map.js
 - Signup.js
 - Workout.js M

- _node-version
- _ruby-version
- .buckconfig
- .eslintrc.js
- .flowconfig
- .gitignore
- .prettierrc.js
- .watchmanconfig
- App.js M
- app.json
- babel.config.js
- Gemfile

OUTLINE

TIMELINE

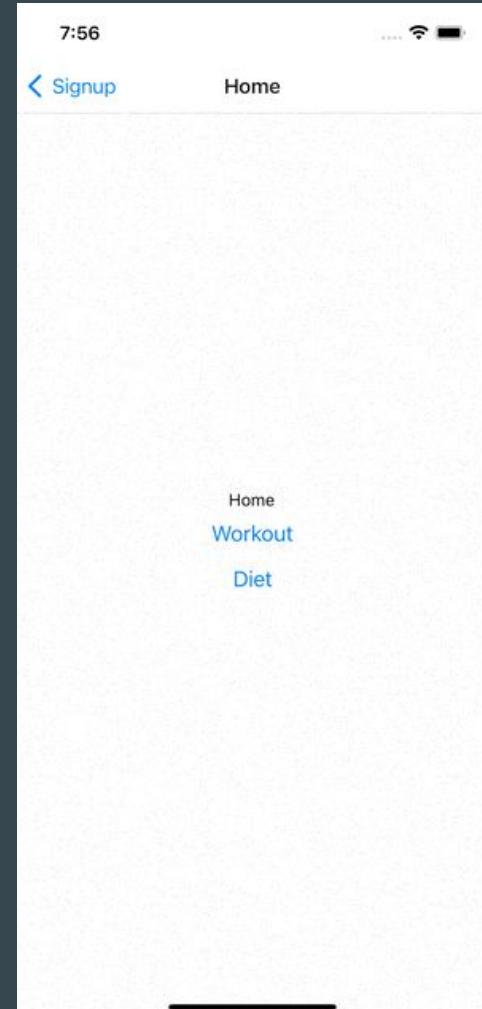
JAVA PROJECTS

App.js

```
1 import React from 'react';
2 import type {Node} from 'react';
3 import {SafeAreaView, StyleSheet, Text, View} from 'react-native';
4 import {NavigationContainer} from '@react-navigation/native';
5 import {createNativeStackNavigator} from '@react-navigation/native-stack';
6 import Home from './Screens/Home';
7 import Workout from './Screens/Workout';
8 import Signup from './Screens/Signup';
9 import Diet from './Screens/Diet';
10 const Stack = createNativeStackNavigator();
11
12 const App: () => Node = () => {
13   return [
14     <NavigationContainer>
15       <Stack.Navigator initialRouteName="Signup">
16         <Stack.Screen name="Home" component={Home} />
17         <Stack.Screen name="Workout" component={Workout} />
18         <Stack.Screen name="Signup" component={Signup} />
19         <Stack.Screen name="Diet" component={Diet} />
20       </Stack.Navigator>
21     ];
22   };
23
24 export default App;
25
```

Ln 17, Col 60 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier

```
return (
  <SafeAreaView style={styles.mainContainer}>
    <TextInput
      multiline
      numberOfLines={14}
      onChangeText={text => handleTextChange(text)}
      style={styles.textInput}
      placeholder={
        'Enter everything you ate today!\n(Press "return" to start a new line)'
      }
      placeholderTextColor="gray"
    />
    <Button title="Send" onPress={() => handleSendButton()} />
  </SafeAreaView>
);
}
```



Diet.js — ReactNativeFundamentals

EXPLORER ... Diet.js U X Home.js M Signup.js App.js M

REACTNATIVEFUNDAMENTALS

- _Xcode.env
- Podfile
- Podfile.lock M
- > node_modules
- Screens
- Diet.js U
- Home.js M
- Map.js
- Signup.js
- Workout.js M
- _node-version
- _ruby-version
- .buckconfig
- .eslintrc.js
- .flowconfig
- .gitignore
- .prettierrc.js
- .watchmanconfig
- App.js M
- app.json
- babel.config.js
- Gemfile

OUTLINE

TIMELINE

JAVA PROJECTS

Diet.js

```
13
14 export default function Diet() {
15   const [dietText, setDietText] = useState();
16   const handleTextChange = text => {
17     setDietText(text);
18   };
19   const handleSendButton = () => {
20     console.log('Sent!');
21     Keyboard.dismiss();
22     fetch('https://ReactNativeFundamentalsServer.egrep6021ad.repl.co/create/', {
23       method: 'POST',
24       headers: {
25         Accept: 'application/json',
26         'Content-Type': 'application/json',
27       },
28       body: JSON.stringify({
29         diet: dietText,
30       }),
31     });
32   };
33
34   return (
35     <SafeAreaView style={styles.mainContainer}>
36       <TextInput
37         multiline
38         numberOfLines={14}
39         onChangeText={text => handleTextChange(text)}
40         style={styles.textInput}
41         placeholder={
42           'Enter everything you ate today!\n(Press "return" to start a new line)'
```

master* ⌂ ⌂ 0 △ 0 ▶ React Native Packager ⌂

Ln 15, Col 3 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier ⌂ ⌂

khalfanibozeman — M...

ozeman/Desktop/Fall22/Mobile-App—Development/presentation/ReactNativeFundamentals/Screens/Diet.js: Invalid shorthand property initializer. (68:19)

```
1  from flask import Flask, request
2  app = Flask(__name__)
3
4  # Endpoint:
5  @app.route('/create/', methods=('GET', 'POST'), strict_slashes=False)
6  def accept_post_request():
7      # Parse JSON data:
8      data = request.get_json()
9      print("The data from a ReactNativeFundamentals App:")
10     print(data)
11     return "[SUCESS]"
12
13  # Webpage endpoint:
14  @app.route('/')
15  def index():
16      return '<h1>React Native Fundamentals</h1>'
17
18  app.run(host='0.0.0.0', port=81)
```

info Reloading app...

BUNDLE ./index.js

LOG Running "ReactNativeFundamentals" with {"rootTag":291, "initialProps":{}}

LOG [RENDERING]: Signup Screen

LOG Current user: Khalfani

LOG Current user's email: Khalfani@gmail.com

LOG [RENDERING]: Home Screen

LOG [RENDERING]: Home Screen

LOG [IOS location enabled]

LOG Longitude: -84.38649

LOG Latitude: 33.75272

LOG Sent!

Line 8 : Col 23 History ⌂

_ Console × Shell × +

```
* Running on http://172.18.0.24:81
Press CTRL+C to quit
172.18.0.1 - - [11/Nov/2022 01:04:59] "GET / HTTP/1.1" 200 -
The data from a ReactNativeFundamentals App:
{'diet': 'Apple\nBread\nOrange'}
172.18.0.1 - - [11/Nov/2022 01:05:33] "POST /create/ HTTP/1.1" 200 -
```

iPhone 14 – iOS 16.1

https://ReactNative

8:06

< Home Diet

Apple
Bread
Orange

Send

Alert API
(<https://reactnative.dev/docs/alert>)



APIs

AccessibilityInfo

Alert

Animated

Animated.Value

Animated.ValueXY

Appearance

AppRegistry

AppState

DevSettings

Dimensions

Easing

InteractionManager

Keyboard

LayoutAnimation

Linking

PanResponder

PixelRatio

Platform

PlatformColor

NEW

Alert

Launches an alert dialog with the specified title and message.

Optionally provide a list of buttons. Tapping any button will fire the respective onPress callback and dismiss the alert. By default, the only button will be an 'OK' button.

This is an API that works both on Android and iOS and can show static alerts. Alert that prompts the user to enter some information is available on iOS only.

Example

Function Component

Class Component

Alert Function Component Example



▲ Expo

```
import React, { useState } from "react";
import { View, StyleSheet, Button, Alert } from "react-native";
```

Download [Expo Go](#) and scan
the QR code to get started.

Diet.js — ReactNativeFundamentals

EXPLORER ... Diet.js U X Home.js M Signup.js App.js M

REACTNATIVEFUNDAMENTALS

Screens > Diet.js

```
11 import {useNavigation} from '@react-navigation/native';
12 import {Alert} from 'react-native';
13
14 export default function Diet() {
15   const navigation = useNavigation();
16   const [dietText, setDietText] = useState();
17   const handleTextChange = text => {
18     setDietText(text);
19   };
20   const handleSendButton = () => {
21     //console.log('Sent!');
22     Keyboard.dismiss();
23     fetch('https://ReactNativeFundamentalsServer.egrep6021ad.repl.co/create/', {
24       method: 'POST',
25       headers: {
26         Accept: 'application/json',
27         'Content-Type': 'application/json',
28       },
29       body: JSON.stringify({
30         diet: dietText,
31       })
32     }).then(
33       Alert.alert(
34         'You saved your food log to the remote server!',
35         `Data Saved:\n${dietText}`,
36         [{text: 'OK', onPress: () => navigation.navigate('Home')}],
37       ),
38     );
39   };
40 }
```

master* ⏪ ⏴ 0 ⏵ 0 ▶ React Native Packager ⏪

Ln 34, Col 33 Spaces: 2 UTF-8 LF JavaScript ✓ Prettier ⏹ ⏹

9:13



Signup

Home

Home

Workout

Diet

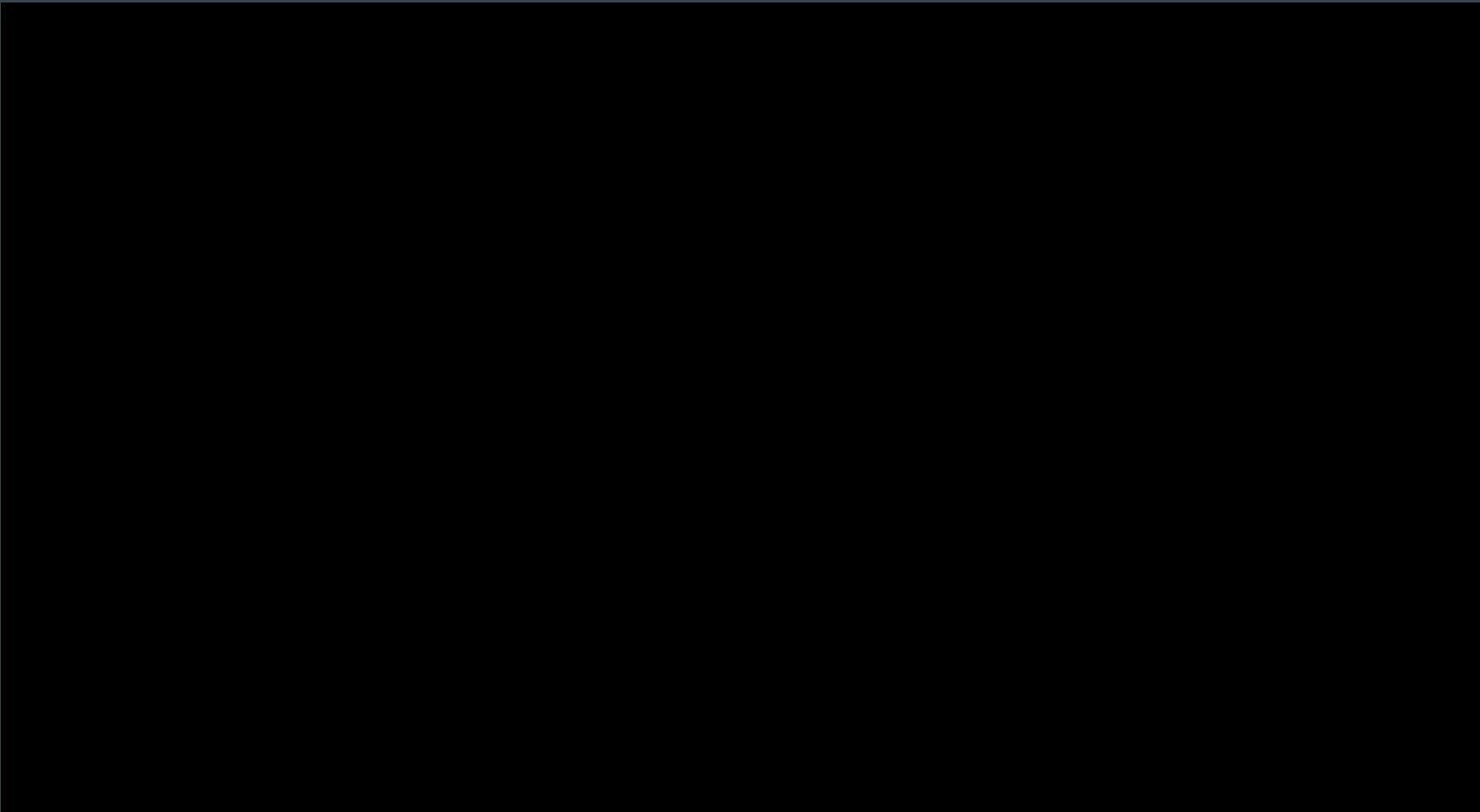
What would you call an Android window that “pops up” when an action is taken?

Dialogs (DatePickerDialog provided by Android)

What is the central object for managing notification-related activities in an iOS app?

“UNUserNotificationCenter”

Finally... A Demo



AndroidManifest.xml — ReactNativeFundamentals

Emulator: Pixel API 30

12:05

12:47

iPhone 14
iOS 16.1

WORKOUT

DIET

Latitude: 37.33045847
`new NativeEventEmitter()` was called with a non-null argument without the required `addListener` method.
`new NativeEventEmitter()` was called with a non-null argument without the required `removeListeners` method.
Running "ReactNativeFundamentals" with {"rootTag":3}
[RENDERING]: Signup Screen
Current user: khal
Current user's email: khal@gsu.edu
[RENDERING]: Home Screen
[RENDERING]: Home Screen
[Android location enabled]
Longitude: -122.083922
Latitude: 37.4220936
info Reloading app...

with identifier Xcode.InterfaceBuilderBuildSupport.Platform
Installing "/Users/khalfanibozeman/Library/Developer/Xcode/Entitlements-ccqootdyiuhcrbrddfbmqpuwkyb/Build/Products/Debug-iphonesimulator/ReactNativeFundamentals.app"
Launching "org.reactjs.native.example.ReactNativeFundamentals"
Successfully launched the app on the simulator

11:53 ReactNativeFundamentals

Event Log Layout Inspector

master

Spaces: 2 LF XML Prettier

Thank you!
(Please “Star” the repository if you find it helpful)