

Spinning Up a React Native Application

Table of Contents

1. System Requirements
2. Installation Instructions
 - a. Node
 - b. Java Development Kit
 - c. Android Studio
 - d. Configuring your Environment Variables
3. Creating a React Native Application
4. Resources

1. System Requirements

The computer I will be using to develop my React Native Application is a custom desktop computer. It has an AMD Ryzen 5 5600X 6-Core processor, 32GB of RAM, and is running on the Windows 11 operating system. It also has a NVMe Solid State Drive with 500GB of storage.

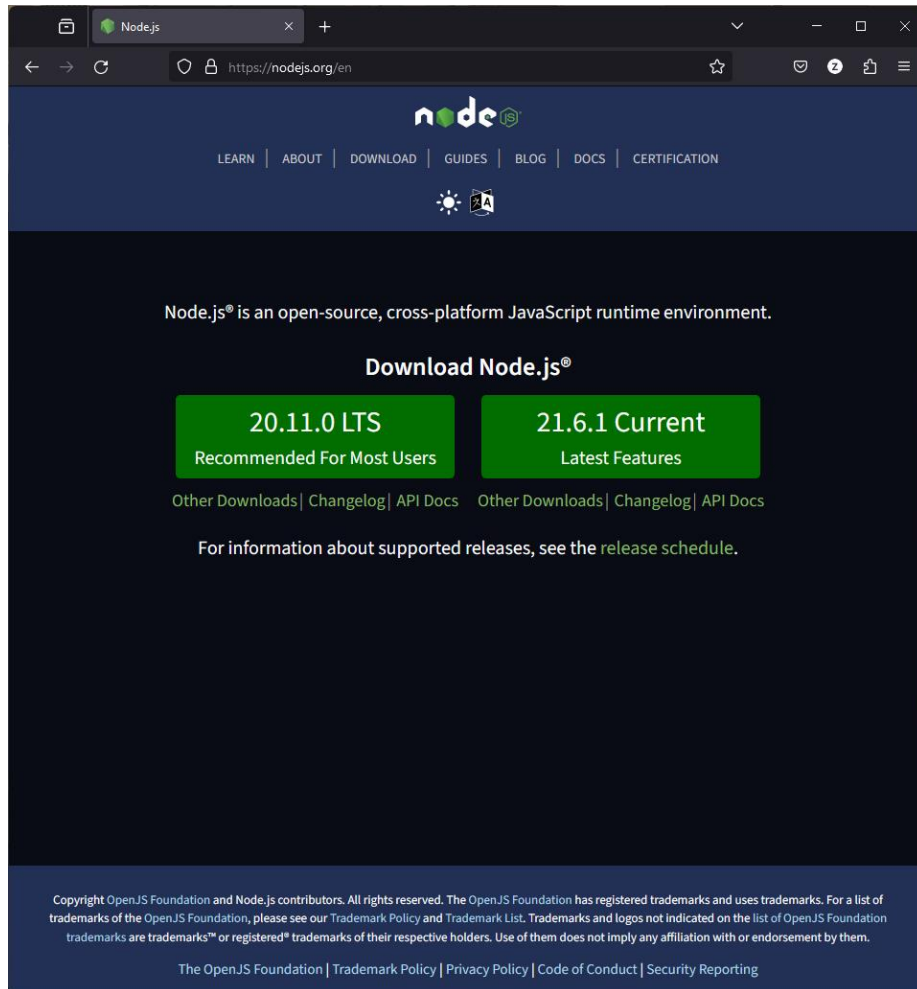
My secondary workplace, which is my laptop, is also being used for React. My laptop is a Lenovo Slim 7, with 16GB of RAM, an AMD Ryzen 7 6800HS processor, and 500GB of storage. It is also running the operating system of Windows 11. Before beginning, VS Code should already be installed as it is what we will be using to change our code once we have created the app.

2. Installation Instructions

To properly use the React Native environment, there are a few tools and frameworks that need to be installed. These tools are Node.js, Expo Go and Android Studio, and a JDK (Java Development Kit).

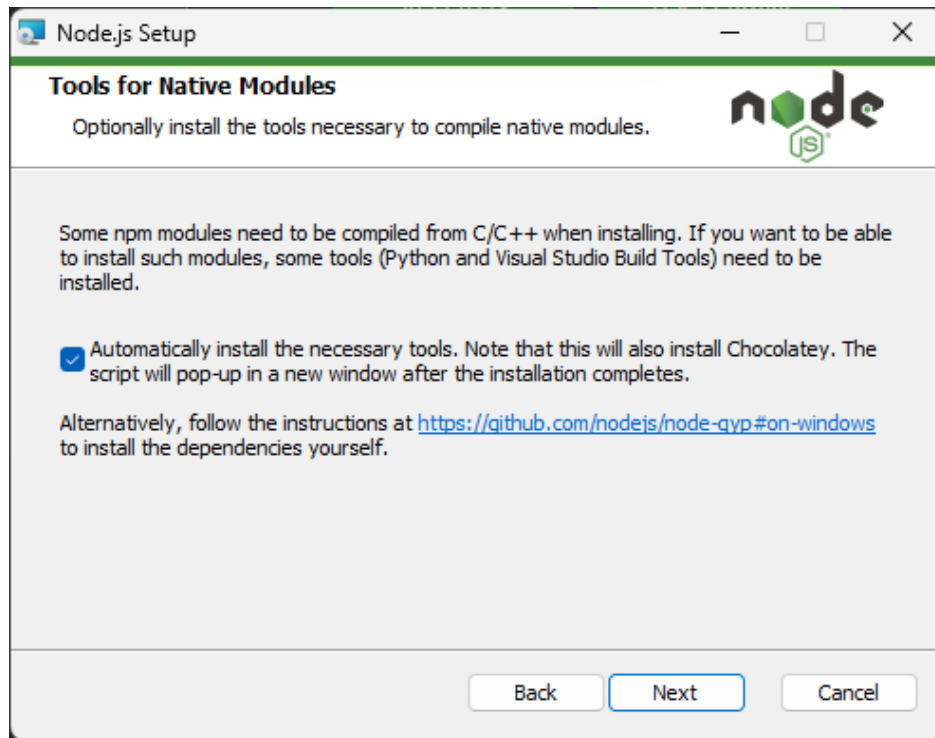
a. Node

To install Node, head to <https://nodejs.org/en>. The most stable version as of January 1, 2023 is currently 20.11.0. Click on version 20.11.0 to begin installation.



[Node homepage](https://nodejs.org/en)

Once you have installed the .msi file, open it to begin installation of Node. Click through the instructions until it prompts you to check 'Automatically install the necessary tools...' This will install Chocolatey which is a package manager for Windows. Click next and wait for Node to finish installing.



Node.js Setup

b. Java Development Kit

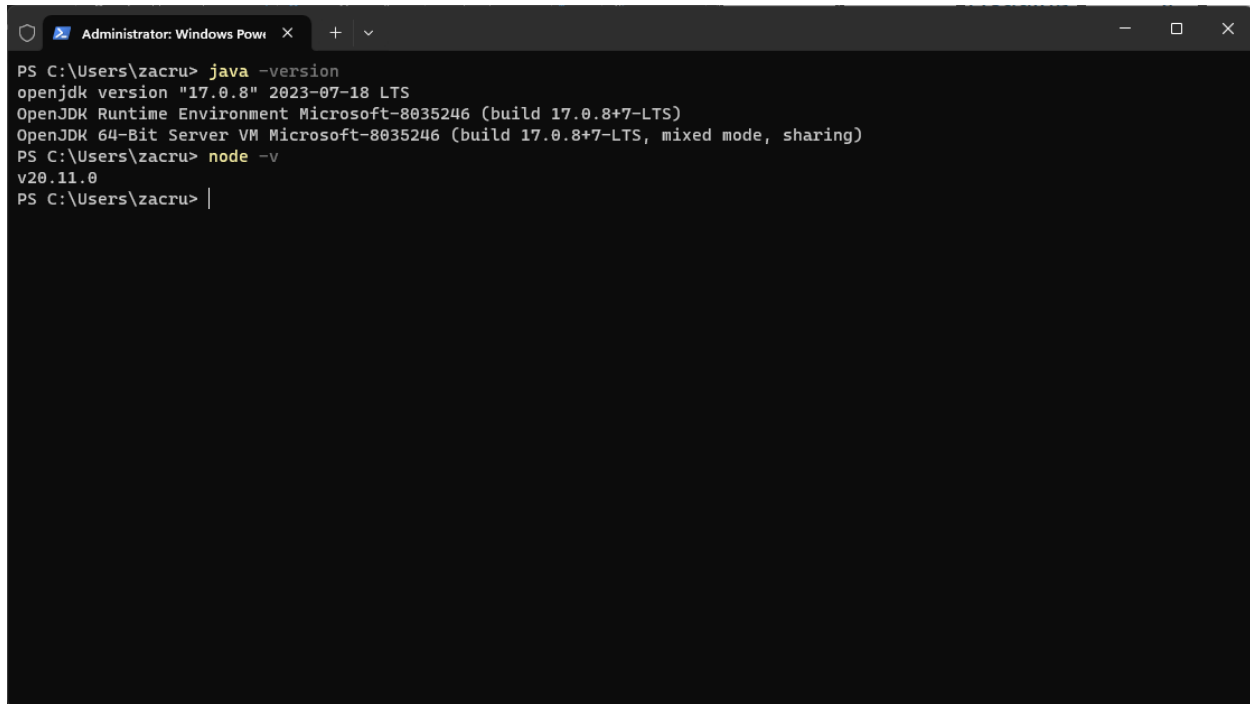
After Node is done installing, you can install the JDK using Chocolatey. Open your command terminal as an Administrator and enter the following command:

```
choco install -y microsoft-openjdk17
```

Node.js and the JDK are now finished installing. You can verify the install is complete by entering the following commands in your command terminal:

```
java -version
```

```
node -v
```

A screenshot of a Windows Command Prompt window titled "Administrator: Windows PowerShell". The window shows the following text:

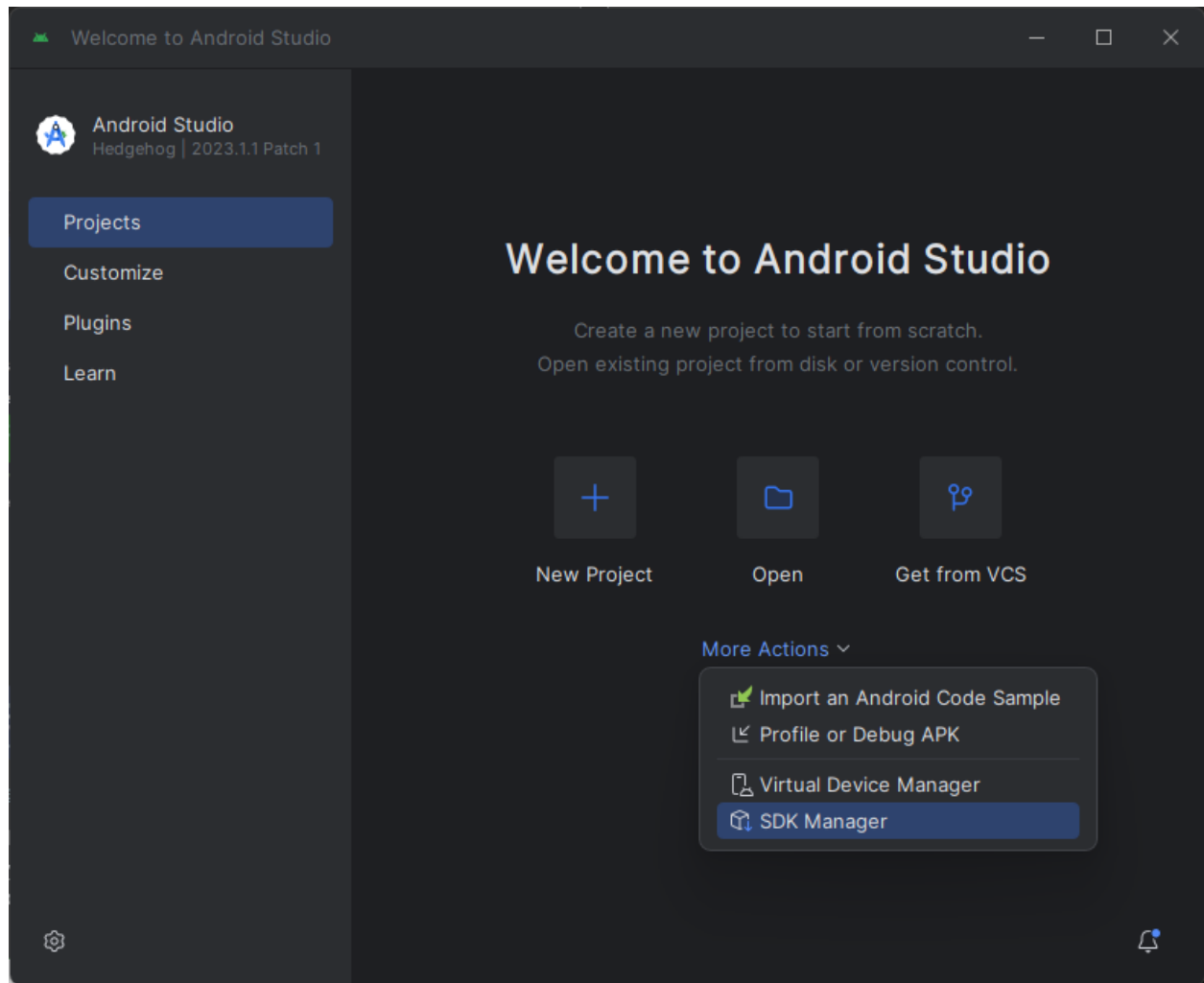
```
PS C:\Users\zacru> java -version
openjdk version "17.0.8" 2023-07-18 LTS
OpenJDK Runtime Environment Microsoft-8035246 (build 17.0.8+7-LTS)
OpenJDK 64-Bit Server VM Microsoft-8035246 (build 17.0.8+7-LTS, mixed mode, sharing)
PS C:\Users\zacru> node -v
v20.11.0
PS C:\Users\zacru> |
```

Command Terminal

You should be able to see the jdk version and the node version we are using. We are now ready to install Android Studio.

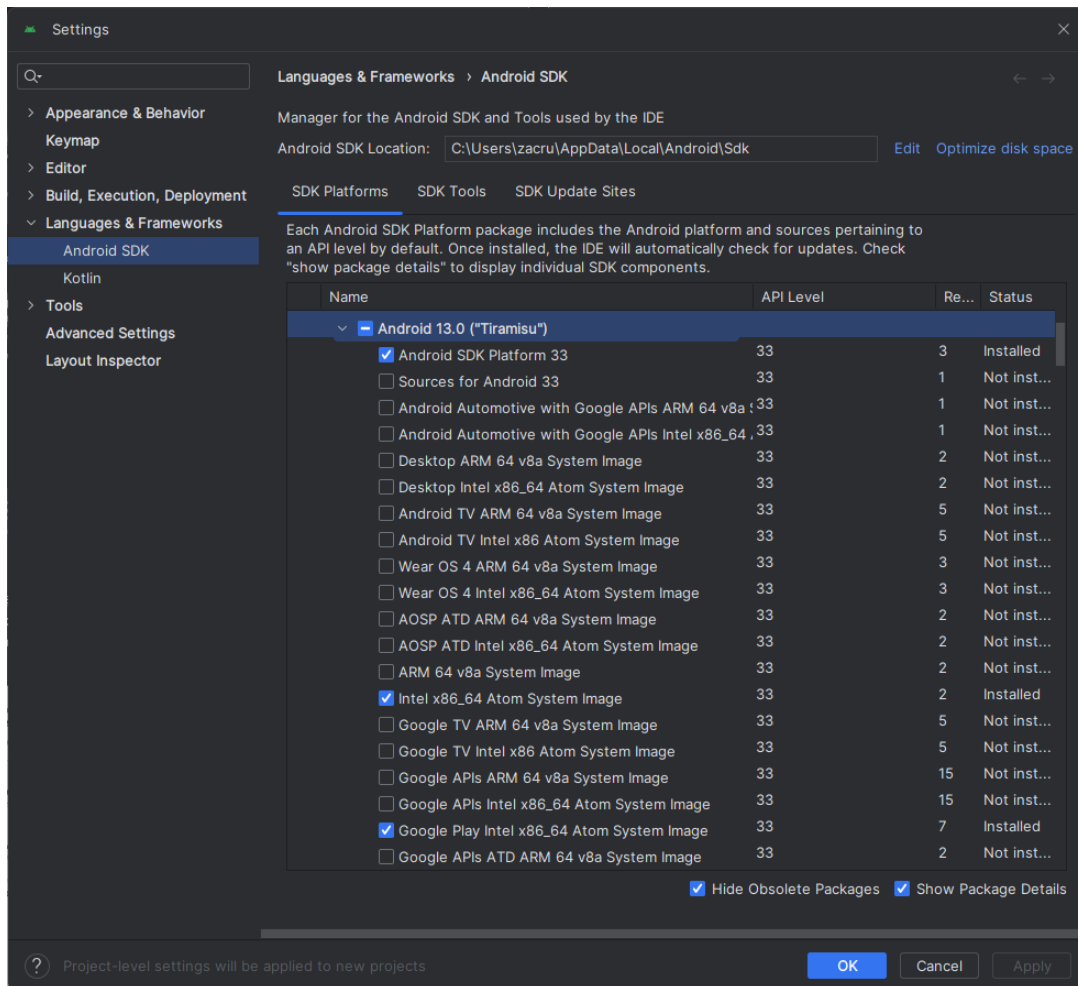
c. Android Studio

Head to <https://developer.android.com/studio/index.html> and download Android Studio Hedgehog. In the installer, make sure you have 'Android Virtual Device' checked and you can click through the rest of the setup. Once Android Studio is installed, ensure you are installing an Android SDK version 13 (Tiramisu). To do this, click on 'More Actions', and then select the SDK Manager.



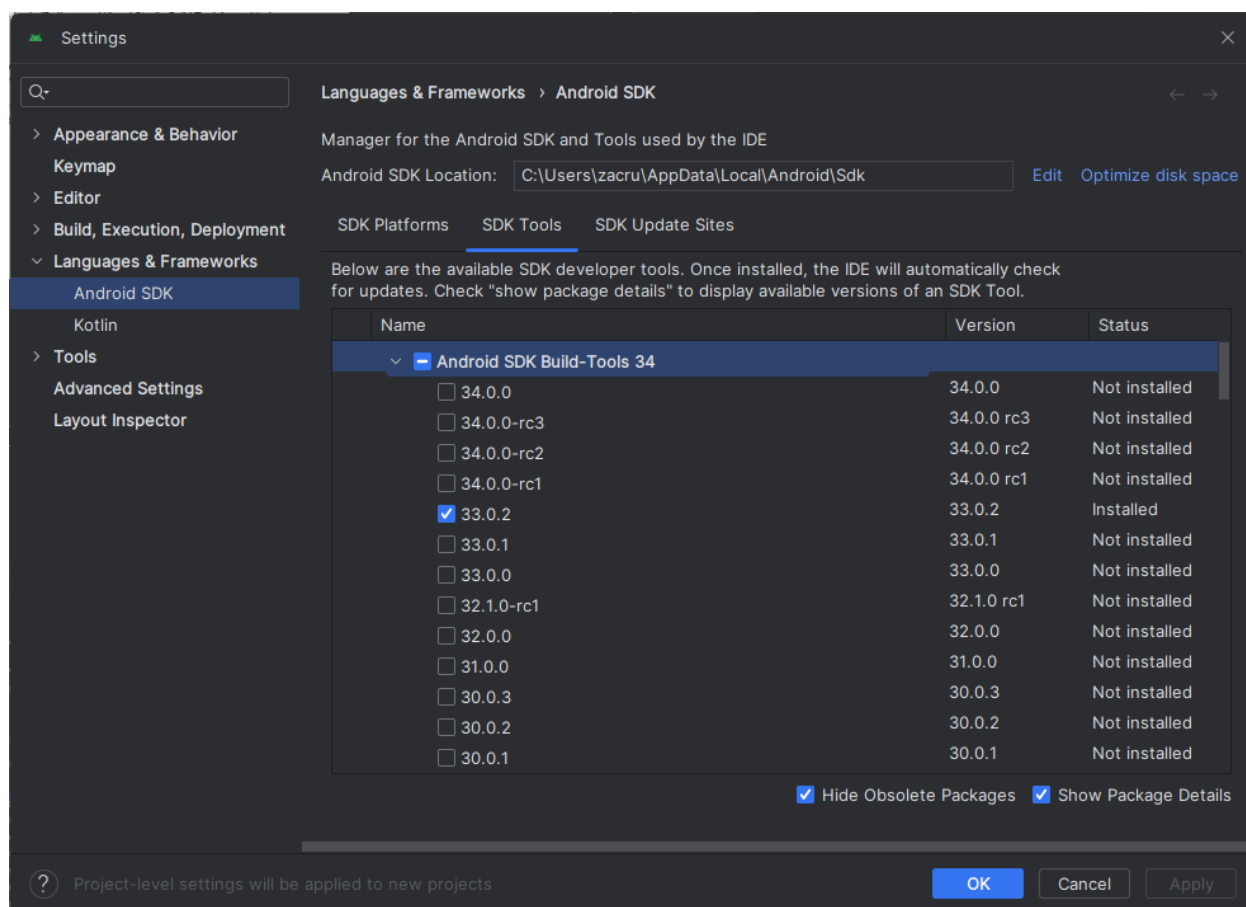
Android Studio SDK Manager

In the SDK Platforms tab, make sure you have the following options checked:



Android Studio SDK Platforms

Lastly, click on the SDK Tools tab and make sure you have version 33.0.2 installed.



Android Studio SDK Tools

d. Configuring your Environment Variables

React native require environment variables to be setup to run properly. Open your Windows Control Panel, click on User Accounts, and then click User Accounts again, and finally select, 'Change my environment variables.' Select 'New...', name the variable ANDROID_HOME and select the folder where your Android SDK is located. After creating the ANDROID_HOME variable, select the environment variable named 'Path' and click 'Edit...' Click New and then Browse and select the folder labeled platform-tools, which is found in the ..\Android\Sdk folder. To verify this has been done properly, open your command terminal and enter the following command:

```
Get-ChildItem -Path Env:\
```

You should see ANDROID_HOME near the top of the list.

```
Administrator: Windows PowerShell
PS C:\Users\zacru> Get-ChildItem -Path Env:\

Name                           Value
----                           -
ALLUSERSPROFILE                C:\ProgramData
ANDROID_HOME                   C:\Users\zacru\AppData\Local\Android\Sdk
APPDATA                        C:\Users\zacru\AppData\Roaming
ChocolateyInstall              C:\ProgramData\chocolatey
ChocolateyLastPathUpdate       133494927248118834
CommonProgramFiles             C:\Program Files\Common Files
CommonProgramFiles(x86)        C:\Program Files (x86)\Common Files
CommonProgramW6432             C:\Program Files\Common Files
COMPUTERNAME                   ZAC-DESKTOP
ComSpec                        C:\WINDOWS\system32\cmd.exe
D3DGEARPATH                    e:\steamlibrary\steamapps\common\iracing\d3dgear
DriverData                     C:\Windows\System32\Drivers\DriverData
HOMEDRIVE                      C:
HOMEPATH                      \Users\zacru
JAVA_HOME                      C:\Program Files\Microsoft\jdk-17.0.8.7-hotspot\
LOCALAPPDATA                   C:\Users\zacru\AppData\Local
LOGONSERVER                     \\ZAC-DESKTOP
NUMBER_OF_PROCESSORS           12
OculusBase                     C:\Program Files\Oculus\
OneDrive                       C:\Users\zacru\OneDrive
OS                              Windows_NT
Path                           C:\Program Files\Microsoft\jdk-17.0.8.7-hotspot\bin;C:\Python312\Scripts\;C:\Python31...
PATHEXT                        .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC;.PY;.PYW;.CPL
PROCESSOR_ARCHITECTURE         AMD64
PROCESSOR_IDENTIFIER           AMD64 Family 25 Model 33 Stepping 0, AuthenticAMD
PROCESSOR_LEVEL                25
```

ANDROID_HOME in PowerShell

We can now open VS Code and create our first React Native Application!

3. Creating a React Native Application

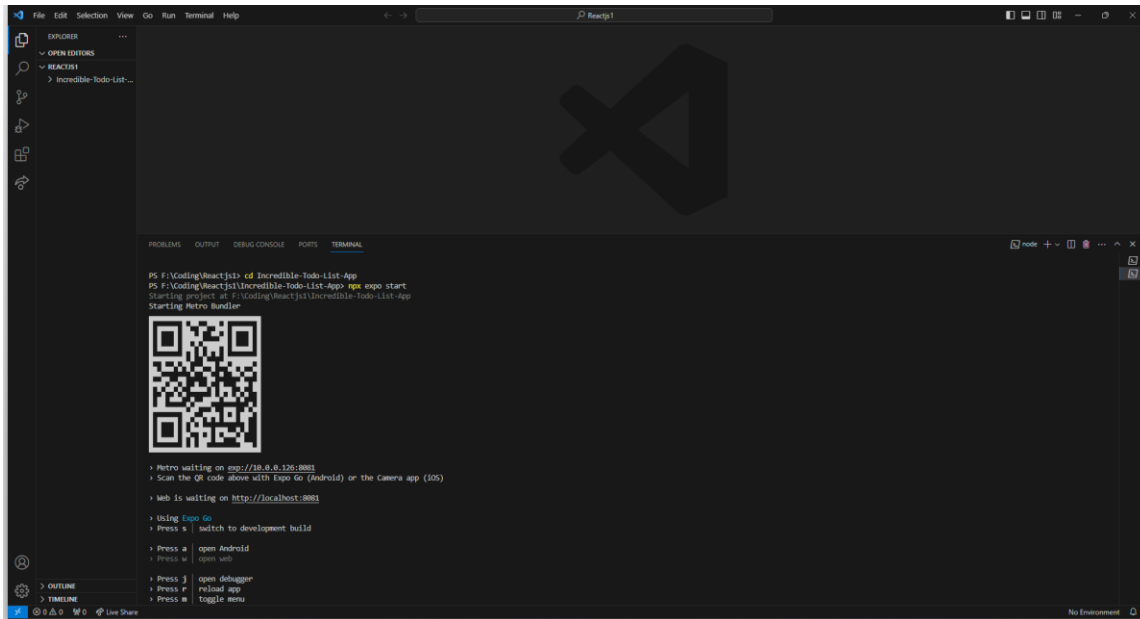
Open VS Code and open a new terminal. In the terminal enter the following commands:

```
npx create-expo-app Incredible-Todo-List-App
```

```
cd Incredible-Todo-List-App
```

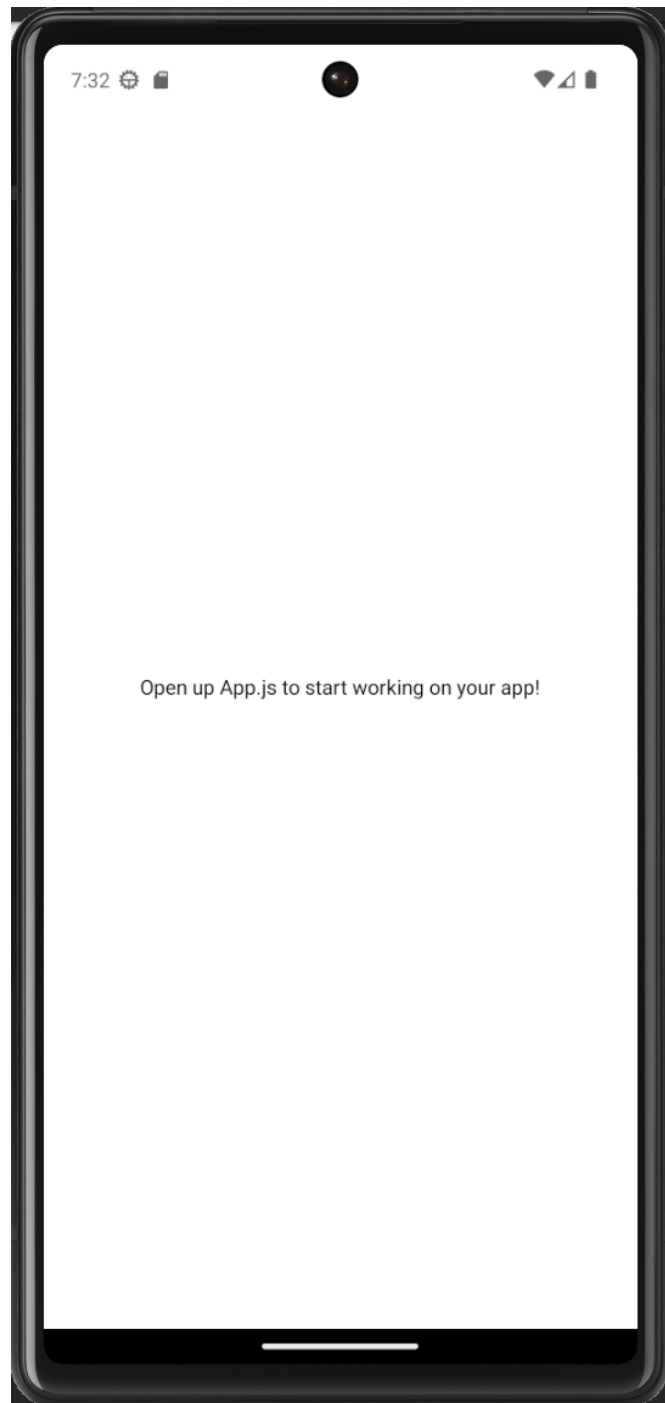
```
npx expo start
```

You should see a QR code that shows your react native app is now up and running.



VS Code terminal showing React is running

Press A on your keyboard to open your Android emulator. Give the emulator some time to boot up and install Expo Go. Expo Go is the app we will be using to create and edit our app.



Expo Go Application

You can now open the App.js file found in the Incredible-Todo-List-App folder we created and begin creating your first React Native application!

4. Resources

1. Official Setup for the React Native Development Environment
 - <https://reactnative.dev/docs/environment-setup?guide=quickstart>
2. Node
 - <https://nodejs.org/en>
3. Android Studio
 - <https://developer.android.com/studio/index.html>
4. React Native Fundamentals
 - <https://reactnative.dev/docs/intro-react>