# React-native

## React-native 初入

### 准备

下载 Android Studio

<https://developer.android.google.cn/studio/>

安装 Expo Client

<https://expo.io/tools>

安装 expo-cli

npm i -g expo-cli

初始化工程

expo init AwesomeProject

cd AwesomeProject

npm start

安装react-native 环境

npm i –g react-native

生成一个react-native

react-native init myapp

### 环境搭建

1. 安装node.js
2. 安装react-native cli

npm i -g react-native-cli

1. 安装Android Studio
2. 生成一个react-native

react-native init myapp

### 安装依赖

必须安装的依赖有：Node、React Native 命令行工具、Python2 以及 JDK 和 Android Studio。

虽然你可以使用任何编辑器来开发应用（编写 js 代码），但你仍然必须安装 Android Studio 来获得编译 Android 应用所需的工具和环境。

#### Node, Python2, JDK

我们建议直接使用搜索引擎搜索下载 Node 、Python2 和[Java SE Development Kit (JDK)](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html)

注意 Node 的版本必须大于等于 10，Python 的版本必须为 2.x（不支持 3.x），而 JDK 的版本必须是 1.8（目前不支持 1.9 及更高版本）。安装完 Node 后建议设置 npm 镜像以加速后面的过程（或使用科学上网工具）。

注意：不要使用 cnpm！cnpm 安装的模块路径比较奇怪，packager 不能正常识别！

npm config set registry https://registry.npm.taobao.org --global

npm config set disturl https://npm.taobao.org/dist --global

#### Yarn、React Native 的命令行工具（react-native-cli）

[Yarn](http://yarnpkg.com/)是 Facebook 提供的替代 npm 的工具，可以加速 node 模块的下载。React Native 的命令行工具用于执行创建、初始化、更新项目、运行打包服务（packager）等任务。

npm install -g yarn react-native-cli

安装完 yarn 后同理也要设置镜像源：

yarn config set registry https://registry.npm.taobao.org --global

yarn config set disturl https://npm.taobao.org/dist --global

#### 1. 安装 Android Studio

[首先下载和安装 Android Studio](https://developer.android.com/studio/index.html)，国内用户可能无法打开官方链接，请自行使用搜索引擎搜索可用的下载链接。安装界面中选择"Custom"选项，确保选中了以下几项：

* Android SDK
* Android SDK Platform
* Performance (Intel ® HAXM) ([AMD 处理器看这里](https://android-developers.googleblog.com/2018/07/android-emulator-amd-processor-hyper-v.html))
* Android Virtual Device

然后点击"Next"来安装选中的组件。

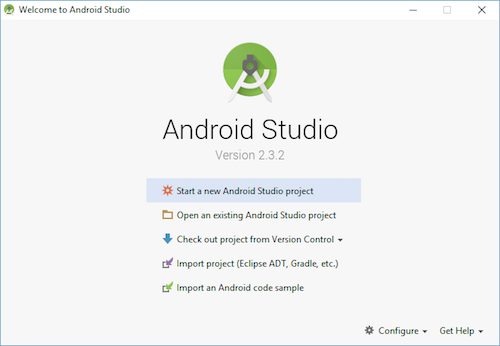
如果选择框是灰的，你也可以先跳过，稍后再来安装这些组件。

安装完成后，看到欢迎界面时，就可以进行下面的操作了。

#### 2. 安装 Android SDK

Android Studio 默认会安装最新版本的 Android SDK。目前编译 React Native 应用需要的是Android 9 (Pie)版本的 SDK（注意 SDK 版本不等于终端系统版本，RN 目前支持 android4.1 以上设备）。你可以在 Android Studio 的 SDK Manager 中选择安装各版本的 SDK。

你可以在 Android Studio 的欢迎界面中找到 SDK Manager。点击"Configure"，然后就能看到"SDK Manager"。

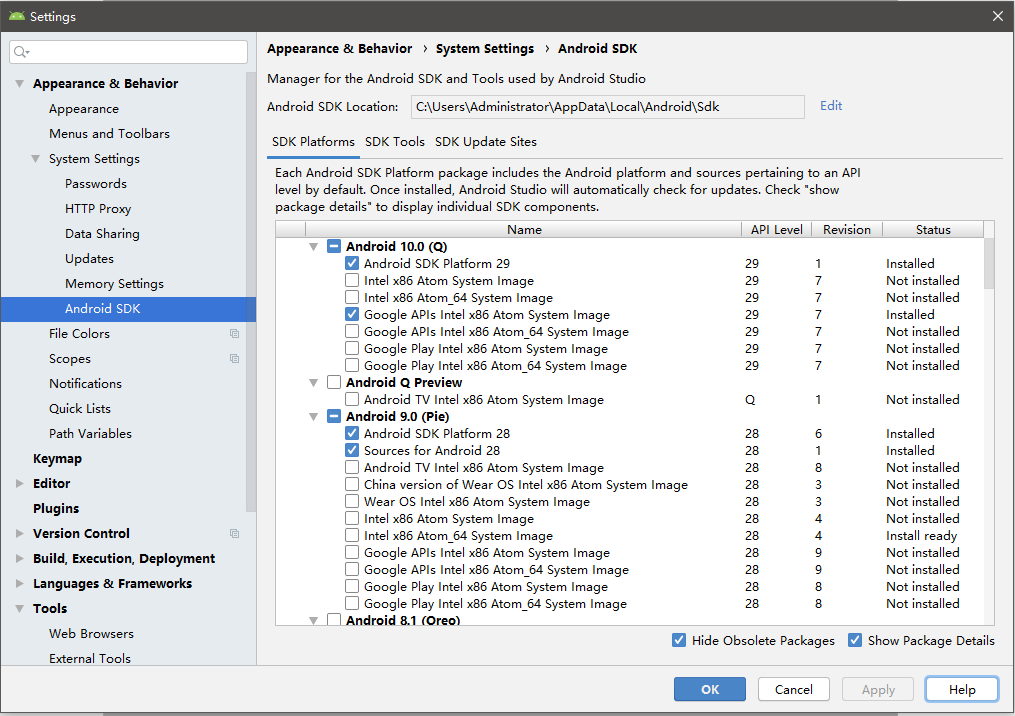


SDK Manager 还可以在 Android Studio 的"Preferences"菜单中找到。具体路径是**Appearance & Behavior** → **System Settings** → **Android SDK**。

在 SDK Manager 中选择"SDK Platforms"选项卡，然后在右下角勾选"Show Package Details"。展开Android 9 (Pie)选项，确保勾选了下面这些组件（重申你必须使用稳定的翻墙工具，否则可能都看不到这个界面）：

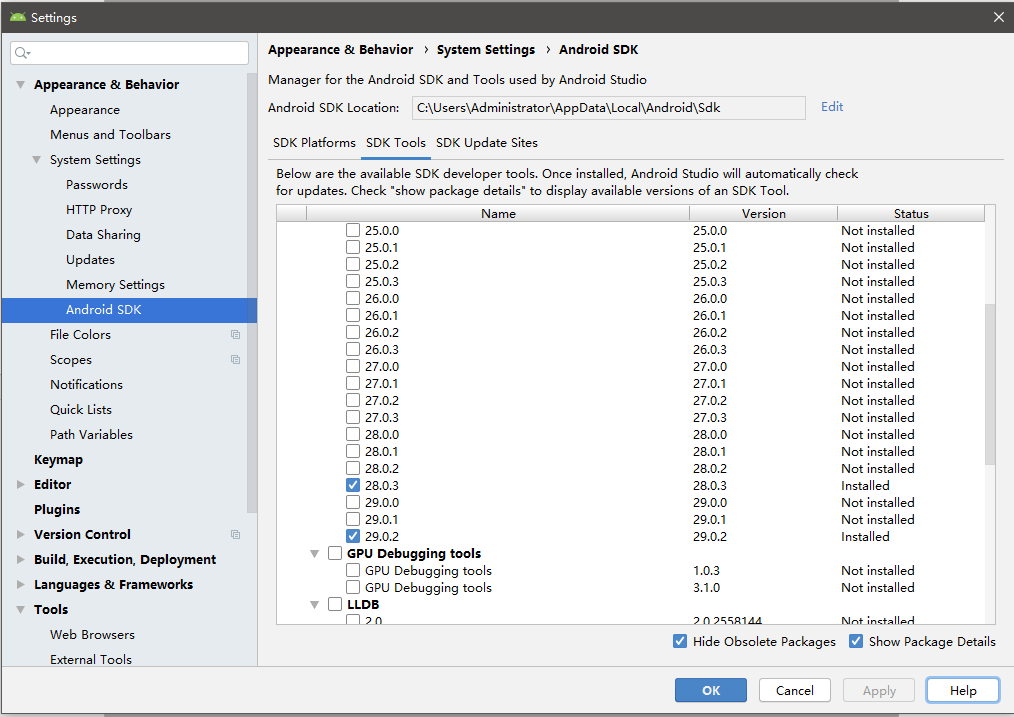
Android SDK Platform 28

Intel x86 Atom\_64 System Image（官方模拟器镜像文件，使用非官方模拟器不需要安装此组件）



然后点击"SDK Tools"选项卡，同样勾中右下角的"Show Package Details"。展开"Android SDK Build-Tools"选项，确保选中了 React Native 所必须的28.0.3版本。你可以同时安装多个其他版本。

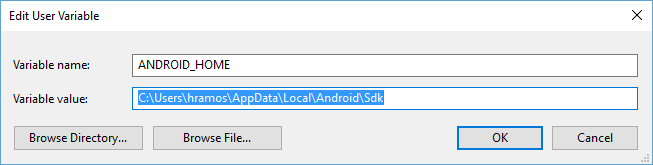
最后点击"Apply"来下载和安装这些组件。



#### 3. 配置 ANDROID\_HOME 环境变量

React Native 需要通过环境变量来了解你的 Android SDK 装在什么路径，从而正常进行编译。

打开控制面板 -> 系统和安全 -> 系统 -> 高级系统设置 -> 高级 -> 环境变量 -> 新建，创建一个名为ANDROID\_HOME的环境变量（系统或用户变量均可），指向你的 Android SDK 所在的目录（具体的路径可能和下图不一致，请自行确认）：



SDK 默认是安装在下面的目录：

c:\Users\你的用户名\AppData\Local\Android\Sdk

你可以在 Android Studio 的"Preferences"菜单中查看 SDK 的真实路径，具体是**Appearance & Behavior** → **System Settings** → **Android SDK**。

你需要关闭现有的命令符提示窗口然后重新打开，这样新的环境变量才能生效。

#### 4. 把 platform-tools 目录添加到环境变量 Path 中

打开控制面板 -> 系统和安全 -> 系统 -> 高级系统设置 -> 高级 -> 环境变量，选中**Path**变量，然后点击**编辑**。点击**新建**然后把 platform-tools 目录路径添加进去。

此目录的默认路径为：

c:\Users\你的用户名\AppData\Local\Android\Sdk\platform-tools

### 创建新项目

使用 React Native 命令行工具来创建一个名为"AwesomeProject"的新项目：

react-native init AwesomeProject

提示：你可以使用--version参数（注意是两个杠）创建指定版本的项目。例如react-native init MyApp --version 0.44.3。注意版本号必须精确到两个小数点。

如果你是想把 React Native 集成到现有的原生项目中，则步骤完全不同，请参考[集成到现有原生应用](https://reactnative.cn/docs/integration-with-existing-apps)。

react-native run-android：

cd AwesomeProject

react-native run-android

译注：建议在run-android成功后再尝试使用 Android Studio 启动。请不要轻易点击 Android Studio 中可能弹出的建议更新项目中某依赖项的建议，否则可能导致无法运行。

### 修改项目

现在你已经成功运行了项目，我们可以开始尝试动手改一改了：

使用你喜欢的文本编辑器打开App.js并随便改上几行

按两下 R 键，或是在开发者菜单中选择 Reload JS，就可以看到你的最新修改。