ReactNative从入门到发疯

目录

一、 前言 1

1. 作者简介 1

2. 本书初衷 1

3. 本书内容简介 1

二、 ReactNative入门必备 2

1. ReactNative介绍 2

1.1 ReactNative 3

1.2 什么是React Native？ 3

1.3 React Native开发注意事项 4

2. ReactNative安装 5

安装Genymotion 7

三、 ReactNative IOS UI/GUI开发 13

1. 你必须要会点FlexBox布局 13

2. React Native常用组件之View 13

3. React Native常用组件之Text 13

4. React Native常用组件之Image 13

5. React Native常用组件之TextInput 13

6. React Native之常用Touchable系列组件 13

四、 ReactNative开发进阶 14

五、 ReactNative 项目实战 14

六、 ReactNative 实战拓展 14

七、 ReactNative”入门到发疯”结尾 14

## 前言

### 作者简介

作者姓名:王衡杰 博客名:香蕉大大(简书) 微信公众号:大大家的iOS说

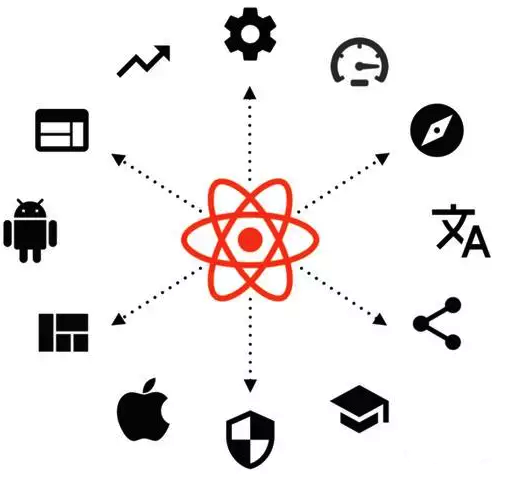
一枚一直在路上的开发者,希望能用自己微小的力量给大家带来一些借鉴以及经验参考!

### 本书初衷

### 本书内容简介

## ReactNative入门必备

### ReactNative介绍



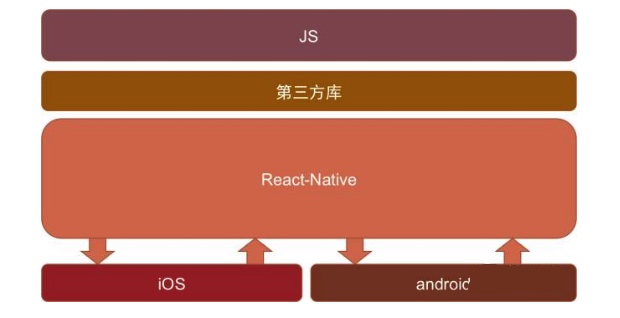
尽管在移动开发中，原生App的开发成本很高，但现阶段基于原生开发仍然是必须的，因为Web的用户体验仍无法超越Native，主要体现在：

* Native的原生控件有更好的体验；
* Native有更好的手势识别；
* Native有更合适的线程模型，尽管Web Worker可以解决一部分问题，但如图像解码、文本渲染仍无法多线程渲染，这影响了Web的流畅性。

### ReactNative

React Native于F8大会开源,在短短不到一年的时间里,它成为手机端必不可少的开发模式之一。 它充分利用了Facebook现有的业务轮子, 其核心设计理念：**既拥有Native的用户体验、又保留React的开发效率**。

目前，React Native基本完成了对多端的支持，实现了真正意义上的面向配置开发: 开发者可以灵活的使用HTML和CSS布局,使用React语法构建组件,实现：**H5, Android, iOS 多端代码的复用**。



此外，使用流畅度和原生的保持在同一层次,这不是我们梦寐以求的开发模式吗?让我们从无到有开启新的学习篇章吧!

### 什么是React Native？

**（一）时下两大移动互联网主流技术**

           BAT等一线互联网公司的插件化，热修改等技术

           React Native技术（**2016年，随着Android版本的稳定，更加火爆**）

**（二）React Native介绍**

          Facebook于2015年9月15日发布**React Native**

          广大开发者可以使用JavaScript和React开发跨平台移动应用

          React Native提倡**组件化**开发: 即提供一个个封装好的组件，组件相互嵌套形成新的组件

（**三）React Native的优势**

**3.1 跨平台开发**

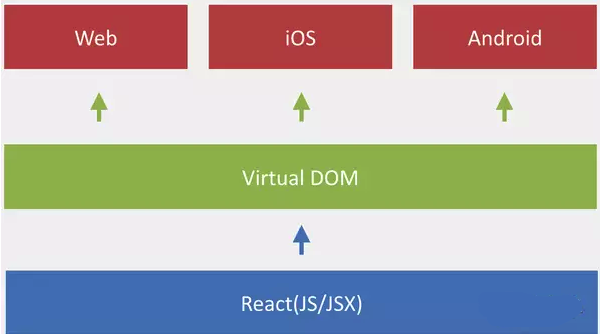
         运用React Native,我们可以使用同一份业务逻辑核心代码来创建原生应用运行在Web端，Android端和iOS端；

**3.2 追求极致的用户体验：实时热部署**

**3.3 learn once,write everywhere** （最具魅力）

        React Native不强求一份原生代码支持多个平台，所以不提“Write once, run anywhere”（Java），提出了“Learn once, write anywhere”。

**React Native开发注意事项**



### React Native开发注意事项

* + 1. 目前react native在iOS上仅支持**ios7**以上，Android仅支持**Android4.1**以上版本:

github地址:  https://github.com/facebook/react-native

       官网文档: http://facebook.github.io/react-native/docs/getting-started.html

* + 1. 由于React Native的版本更新速度很快，如果没有深厚的JavaScript基础，建议选择:

**a、功能适中，交互一般，不需要特别多的系统原生支持；**

**b、对于部分复杂的应用，可以考虑原生+React Native混合开发**

### ReactNative安装

**(一)、环境需求**

**1.1  安装Homebrew**

       Homebrew是OS X的套件(包)管理器，我们可以通过它获取并且安装很多组件

**安装方式:**

      ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

**1.2  安装npm 和 Node.js**

   Node.js最好安装**4.0**及其以上更高版本，node安装成功后npm自动也就有了，直接下载安装Node.js，网址：https://nodejs.org/en/download/ （资料中已有

**1.3  安装WatchMan**

该插件用于监控bug文件和文件变化 ，并且可以触发指定的操作

**安装方式**

      brew install watchman



      注意：在Max OS X 10.11（El Capitan)版本中，homebrew在安装软件时可能会碰到/usr/local目录不可写的权限问题。可以使用下面的命令修复：

sudo chown -R `whoami` /usr/local

**1.3  安装Flow**

       flow是一个 JavaScript 的静态类型检查器，建议安装它，以方便找出代码中可能存在的类型错误

安装方式

brew install flow



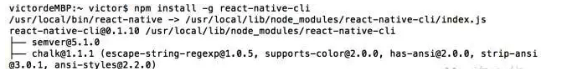
**(注意：如果提示command not found,请加上sudo获得最高权限)**

**(二)、React Native安装**

**2.1 安装React Native**

        npm install -g react-native-cli

**2.2 安装截图如下:**



**(三)、 ios开发环境需求**

Xcode 7 及其以上更高版本

**(四)、 Android开发环境需求**

**安装最新版的JDK:**

      下载安装地址：http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html （资料中已经存放）

      可以安装Android studio省略以下步骤，**建议安装Android studio**。

### ****安装Genymotion****

       Genymotion是一个第三方模拟器，它比Google官方的模拟器更易设置且性能更好。但是，它只针对个人用户免费。

1）下载并安装Genymotion

      https://www.genymotion.com/

2）打开Genymotion,如果你尚未安装VirtualBox,它有可能会提示你安装

3）创建一个模拟器并启动

4）按下⌘+M可以打开开发者菜单(在安装并启动了React Native应用之后)

**(五)、 React Native的第一个应用**

**5.1 执行命令,生成一个工程**

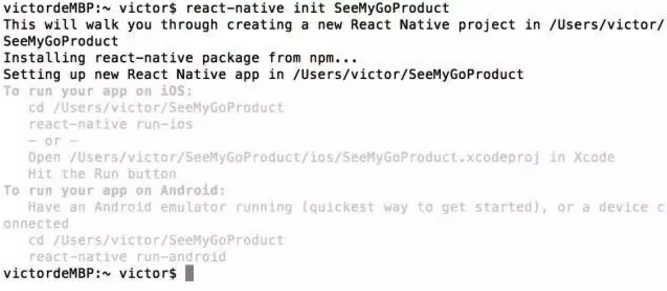
     react-native init 项目名称

      注意:由于众所周知的网络原因，需要等待一段时间（具体视网络情况而定）。react-native命令行从npm官方源拖代码时会遇上麻烦，可以将npm仓库源替换为国内镜像：

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

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

**5.2  运行截图**



**5.3 目录结构截图**

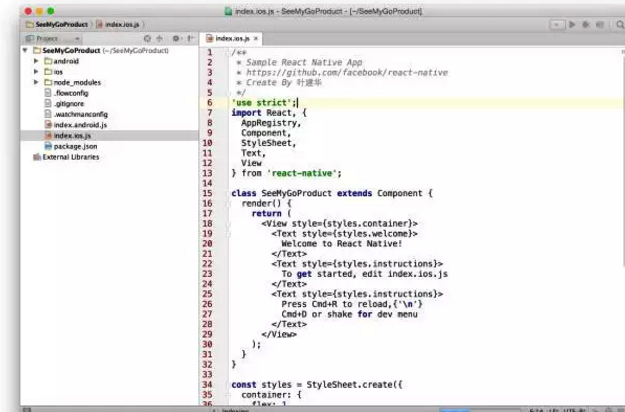
**目录结构分析:**

a）默认生成**android**和**ios**两个平台的原生项目；

b）其中，**index.android.js**和**index.ios.js**文件为Android和iOS的空壳应用文件；

c）此外，**node\_modules**文件夹，是为**Node.js**存放和管理npm包资源，也包含React Native框架文件。

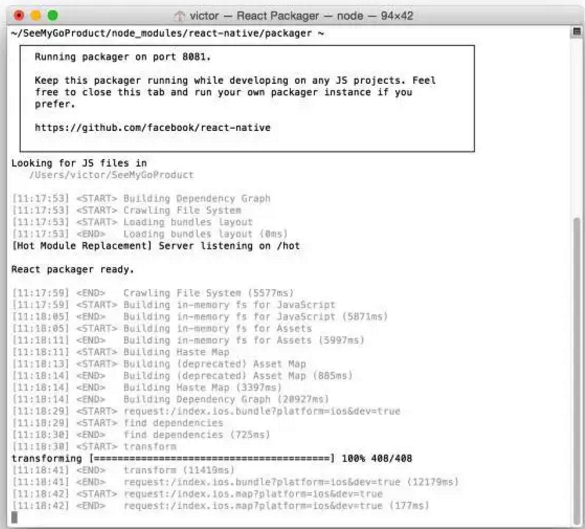
**查看index.ios.js中的代码:**



**(六)、运行工程文件**

      不管是 iOS 还是 Android，在开发调试阶段，都需要在 Mac 上启动一个 HTTP 服务，称为**Debug Server**，默认运行在 **8081** 端口，APP 通 Debug Server 加载 js。

**6.1 打开Xcode，运行你的第一个React Native创建的iOS应用**



初始化成功



启动页面

**6.2 把React Native创建的应用跑在Android上**

a) 命令行执行cd SeeMyGoProduct,路径切换到项目主目录

b) 命令行执行react-native run-android进行加载运行android 应用。

c) 使用编辑器进行打开和修改index.android.js文件，接着通过菜单按钮选择Reload JS来进行刷新修改

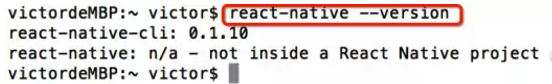
**(七)、管理React Native库的版本**

       在开发中，会经常的去控制React Native的版本库，得以适配各种条件下的开发，那该如何查看、控制ReactNative的版本呢？

**7.1 查看本地的React Native的版本**

  命令行输入

   react-native –version



**7.2 更新本地的React Native的版本**

    命令行输入

    npm update -g react-native-cli

**7.3 查询react-native的npm包最新版本**

NPM的全称是Node Package Manager ，是一个NodeJS包管理和分发工具，已经成为了非官方的发布Node模块（包）的标准。

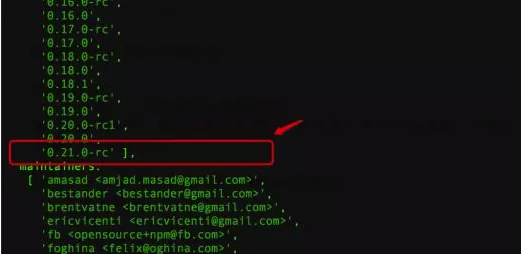
    npm包地址 ：

     https://www.npmjs.com/package/react-native

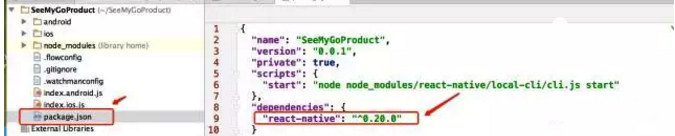
    命令行查询

      npm info react-native

    查询效果



 项目中查看



**7.4 升级或者降级npm包的版本**

    npm install --save react-native@0.18

**7.5 更新项目templates文件（可选）**

       新的npm包会包含更新在运行react-native init命令生成的一些动态文件，例如init创建项目的时候会生成iOS和Android的子项目，我们可以通过以下的命令进行获取最新的代码

     命令行查询

      react-native upgrade

**(八)、WebStom设置React Native代码提示**

**8.1  从gitHub上下载xml插件**

 git clone https://github.com/virtoolswebplayer/ReactNative-LiveTemplate

**8.2  安装**

将ReactNative.xml复制到 ~/Library/Preferences/WebStorm11/templates ，然后重启 WebStrom

请注意,webStorm中可能没有templates 安全起见就自己创建一个templates 文件夹就可以

## ReactNative IOS UI/GUI开发

### 你必须要会点FlexBox布局

在H5中，界面的搭建都是采用CSS的布局，基于**盒子模型**，依赖 **display属性 ， position属性， float属性**。但对于那些特殊布局非常不方便，比如，垂直居中。

一种全新的针对web和移动开发布局应运而生：**Flex布局**，可以简便、完整、响应式地实现各种页面布局。目前，它已经得到了所有浏览器的支持，这意味着，现在就能很安全地使用这项功能。



### React Native常用组件之View

### React Native常用组件之Text

### React Native常用组件之Image

### React Native常用组件之TextInput

### React Native之常用Touchable系列组件

## ReactNative开发进阶

## ReactNative 项目实战

## ReactNative 实战拓展

## ReactNative”入门到发疯”结尾