**APP开发流程**

APP开发采用ionic2进行开发，ionic官网http://ionicframework.com/docs/。

开发软件采用Visual Studio,安装包百度下载即可。

使用ionic进行app开发大致流程如下：

新加内容

# 配置ionic环境

<https://120.27.99.96/svn/JTYS/trunk/02_Code/Ionic技术文档/IONIC开发环境配置.html>

其中包括：

## 1、node简介：

1)Node 是一个服务器端 JavaScript 解释器，它将改变服务器应该如何工作的概念。它的目标是帮助程序员构建高度可伸缩的应用程序，编写能够处理数万条同时连接到一个（只有一个）物理机的连接代码。

2)详见：<http://www.ibm.com/developerworks/cn/opensource/os-nodejs/>，或查阅相关资料。

## 2、node及npm安装：

1）安装node.js:

进入官网：<https://nodejs.org/en/>，选择适合版本安装，node安装自带npm（node package manager）。

1. 安装成功检查CLI（command-line interface）：  
   node -v：查看当前node版本号  
   npm -v：查看当前npm版本号

配置环境变量：在path中有nodejs安装的路径（一般不需手动配置）

3)使用npm安装插件 npm install <name> [-g] [--save-dev]：  
<name>：node插件名称。  
-g：全局安装，可以通过命令行在任何地方调用它，本地安装将安装在定位目录的node\_modules文件夹下，通过require()调用。  
--save：将保存配置信息至package.json。  
-dev：保存至package.json的devDependencies节点

4)cnpm安装：  
CLI：npm install cnpm -g --registry=https://registry.npm.taobao.org  
因为npm安装插件是从国外服务器下载，受网络影响大，可能出现异常，如果npm的服务器在中国就好了，所以我们乐于分享的干了这事。 cnpm跟npm用法完全一致，只是在执行命令时将npm改为cnpm（以下操作将以cnpm代替npm）。

5)node的升级：

## 3、cordova的安装：

1)cli：npm（cnpm） install -g cordova

2)检查是否安装成功：cordova -v

## 4、ionic的安装：

1)npm install -g ionic

2)检查：ionic -v

## 5、ionic官方文档解读：

1)百度‘ionic’

2)点击ionic Documentation，进入官网

## 6、安装安卓开发平台：

1)推荐安装android studio，集成android sdk的安装。

2)配置android sdk的环境变量，在path添加sdk的路径。

（1）ANDROID\_SDK\_HOME，变量值为你的SDK安装路径；

（2）%ANDROID\_SDK\_HOME%\platform-tools;%ANDROID\_SDK\_HOME%\tools添加到Path环境变量中

（3）检测是否配置成功。打开命令行窗口，输入adb。

## 7、IOS开发平台：

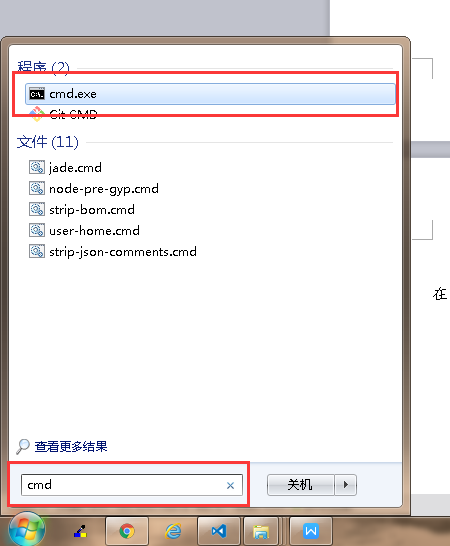
1)安装xcode。

2)在命令行执行ionic info根据提示安装缺少的环境。

# 二、ionic 框架的基本搭建

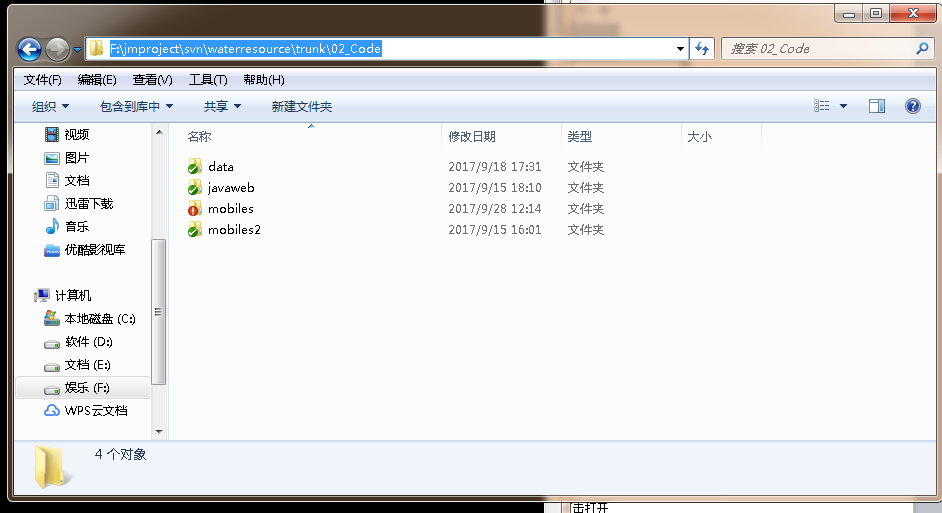
## 通过cmd进入文件路径

1）打开命令窗口

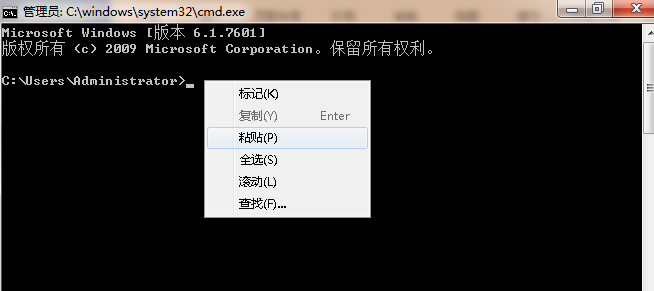


如图所示在任务栏输入cmd，选择cmd.exe点击打开

2）打开要存放项目的文件位置复制路径



3）在命令行中输入”cd “点击右键选择粘贴，粘贴刚刚复制的文件路径



这里cd表示进入某个路径中，



4）点击回车键，再输入F:，再点击回车键，F等表示你当前存放文件的盘。

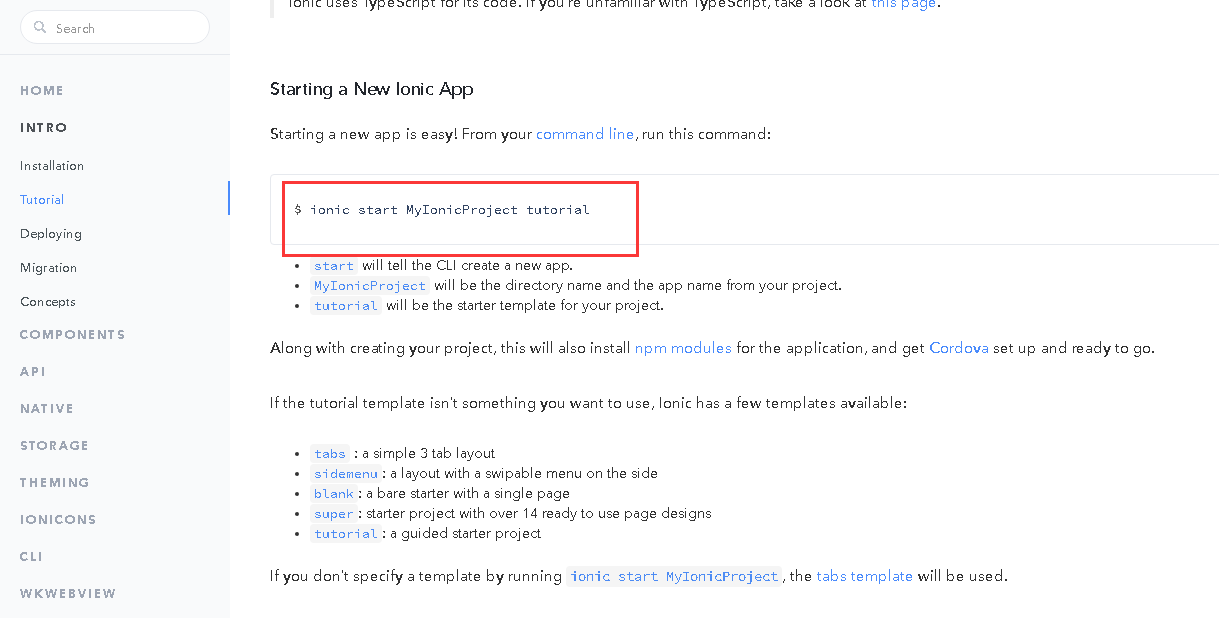


此时我们已经进入自己要把项目存放的目录下。

## 项目的创建

1. 打开ionic官网进入项目创建demo示例依照步骤运行命令行创建项目：

官网地址：[http://ionicframework.com/docs//intro/tutorial/](http://ionicframework.com/docs/intro/tutorial/)



ionic start 项目名称 tabs

其中项目名称表示你要给自己项目起的名字，tabs则是你的项目框架的样式

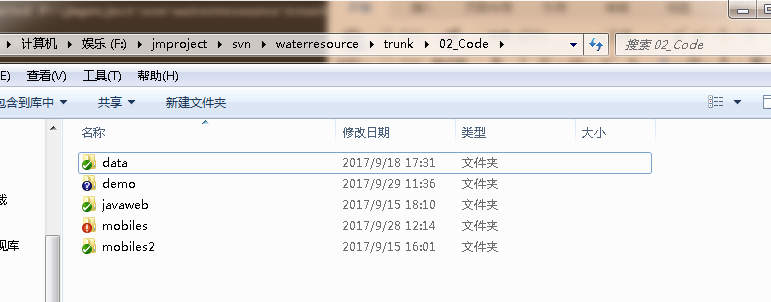
这种就是tabs框架，也可选择其他样式



运行到提示要加载npm资源要等待时直接ctrl+c键进行关闭



这个时候我们就会在我们的文件路径中看到一个demo的文件夹已经被创建成功。

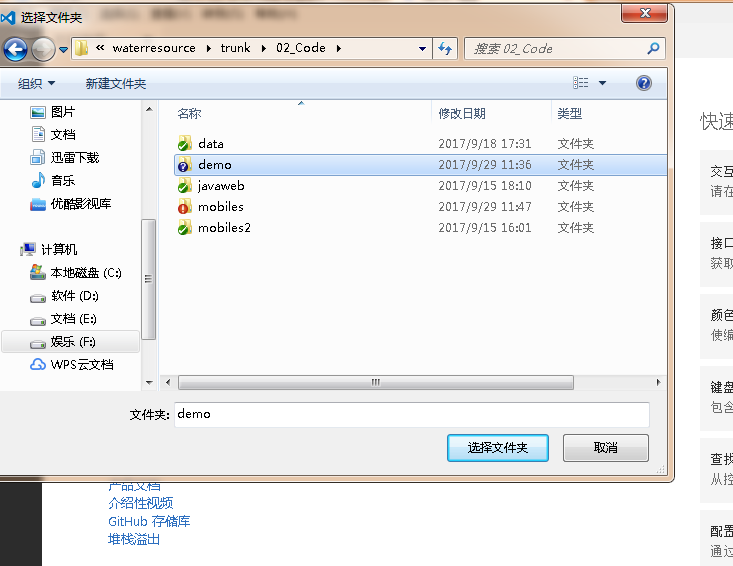


## visual studio 配置

1. visual studio插件的安装

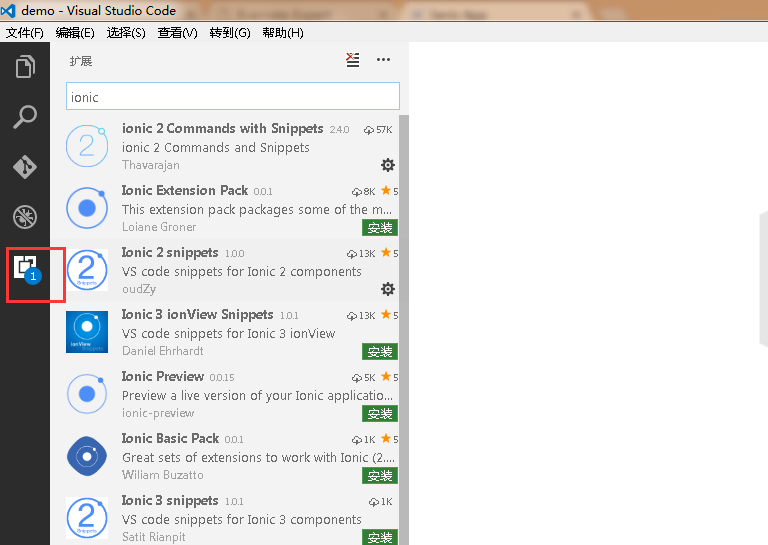
打开visual studio 我们通过visual studio打开我们刚刚创建的demo文件夹



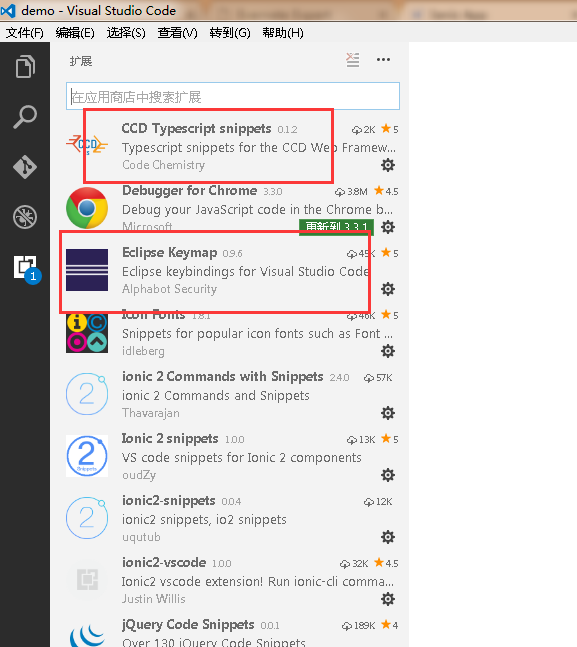


打开文件夹后，我们选择visual studio第四项扩展功能，

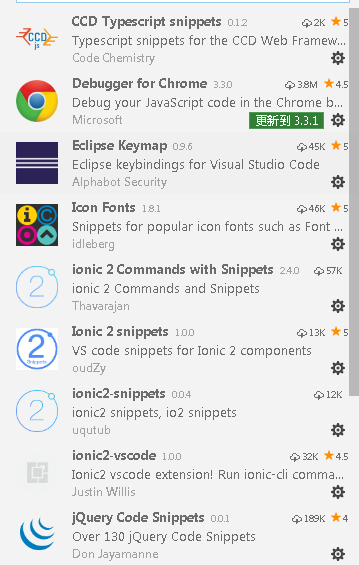
输入ionic将所有ionic2相关的插件点击安装，



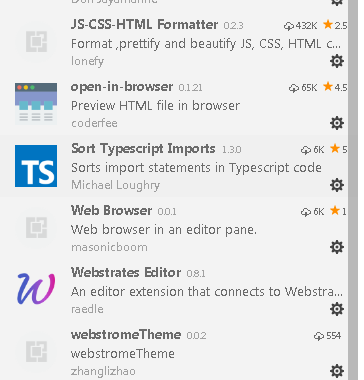
安装typescript、eclipse keymap、js-css-html formatter等插件



目前我安装的插件有：

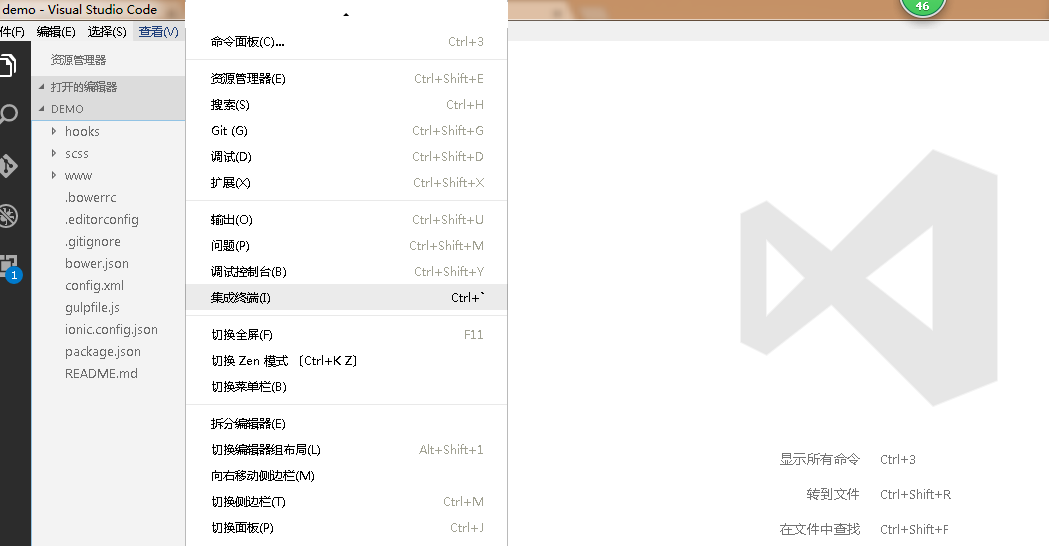


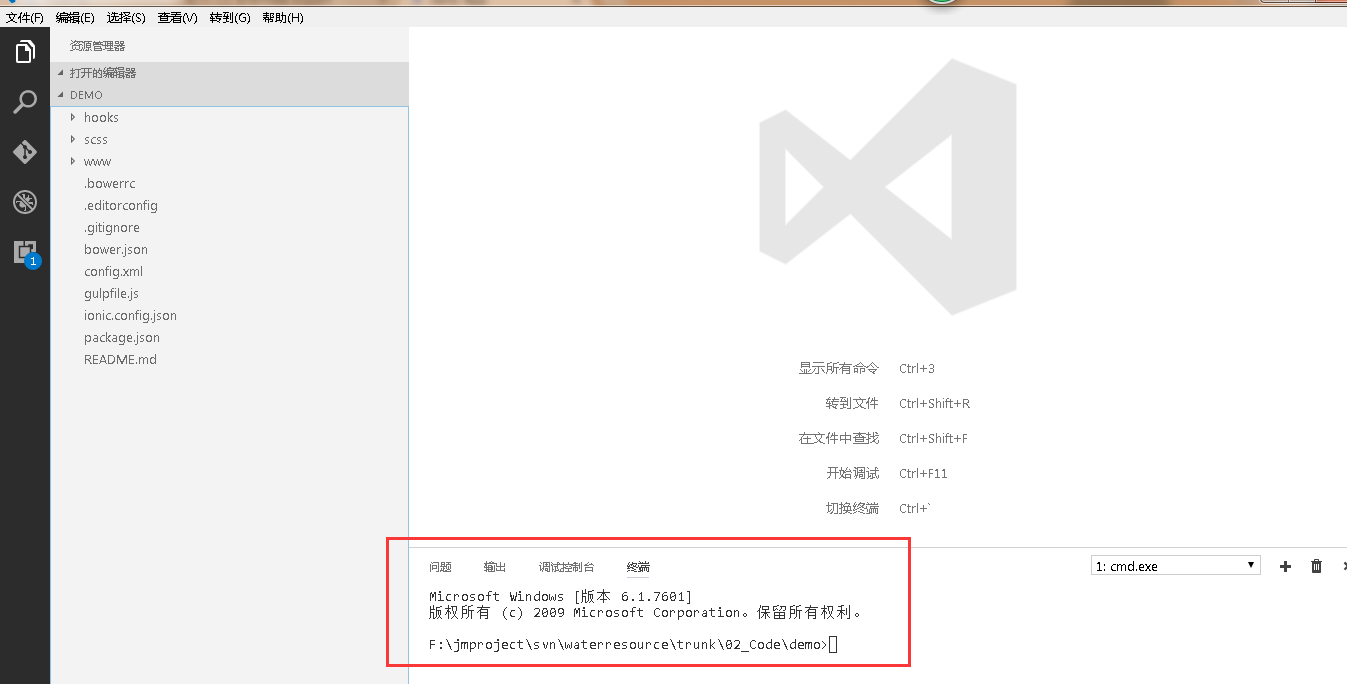
1. 安装完插件后、我们点击提示的重新加载选项，让visual studio重新打开，此时所有插件已经可以正常使用。



## 项目资源的添加

插件安装完成后点击查看，选择集成终端





我们打开集成终端后就可以在这里输入命令来加载项目资源。

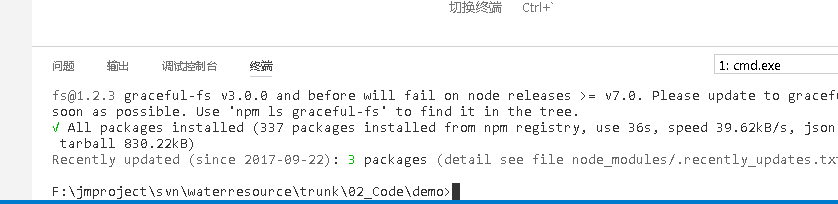
1. 输入cnpm install



加载所有相关ionic资源

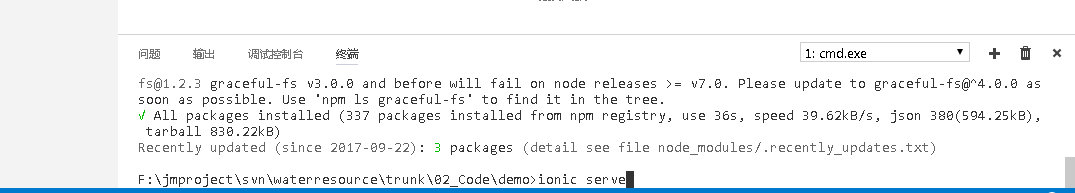


等待加载完成后，项目所需所有资源就加载完成了。

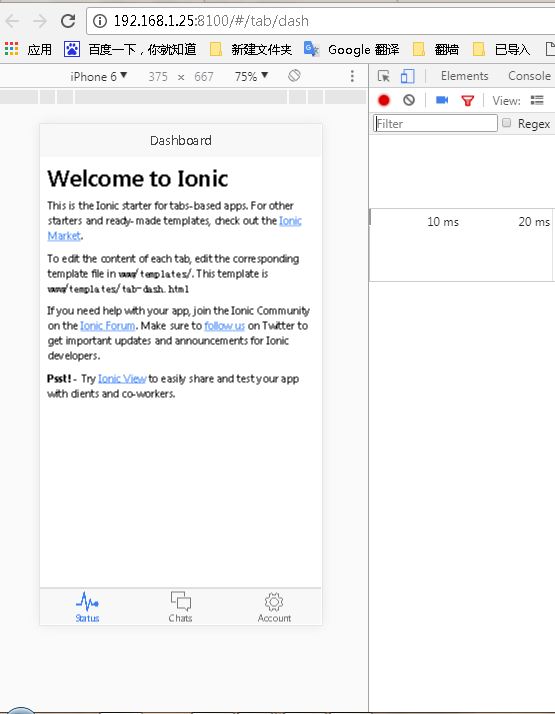


1. 项目浏览

当加载完成后输入ionic serve，项目自动在浏览器中运行



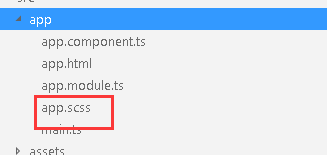
将浏览器放在手机模式下进行查看即可看到现在项目的框架已经基本完成



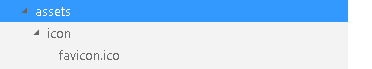
# 项目文件结构



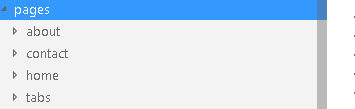
Src文件夹里面为我们要开发的文件存放位置



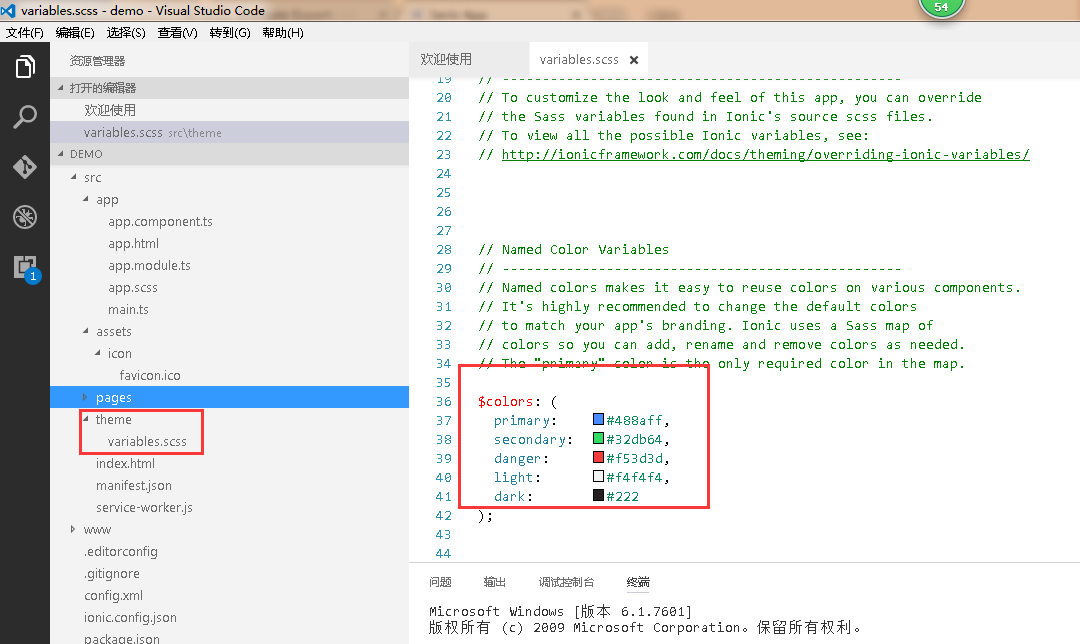
App.scss为公共css文件夹



存放所有自己引入的图片字体等资源



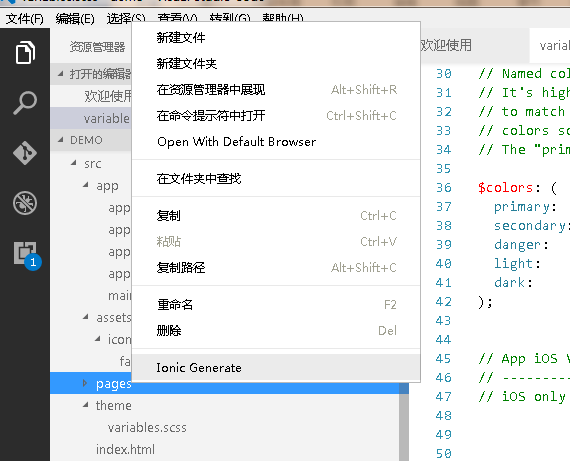
Pages存放所有页面



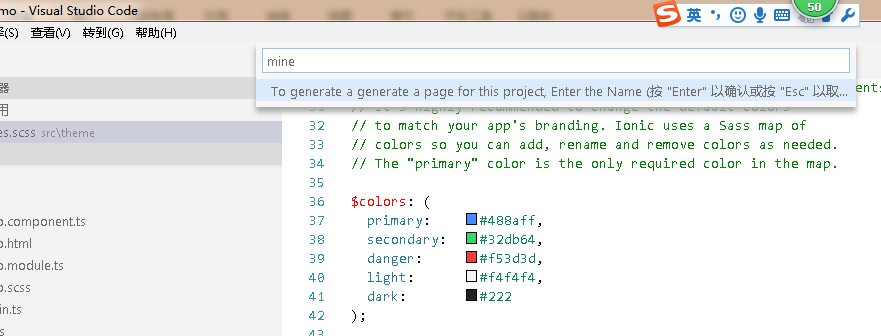
主题配置

## 页面的添加

选中pages文件夹，右键点击ionic generate

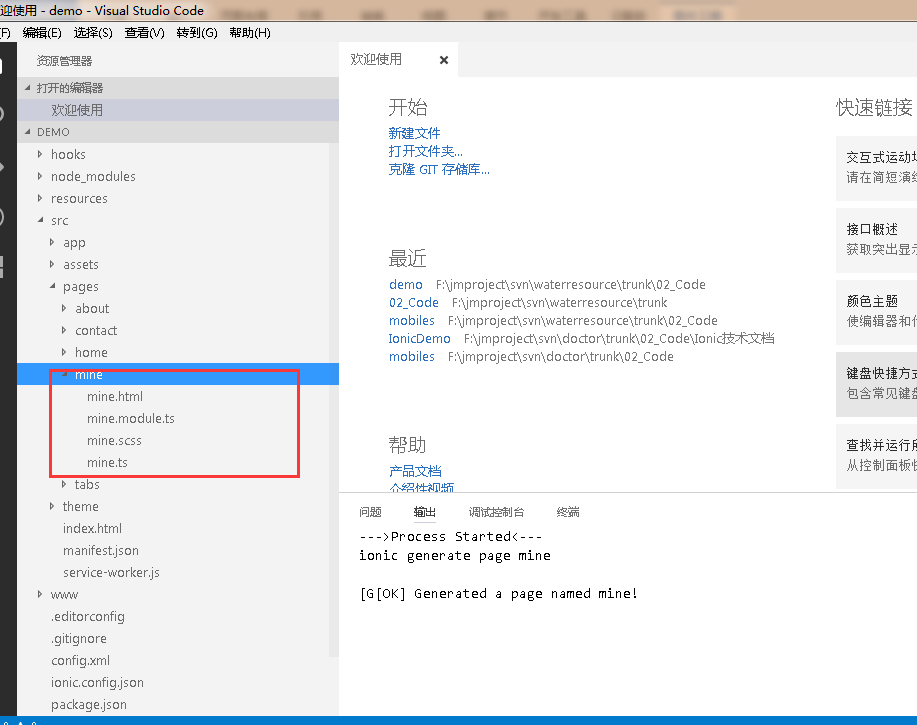


在弹出框中输入要创建的页面名称



回车后，mine这个文件就会被创建在pages文件中





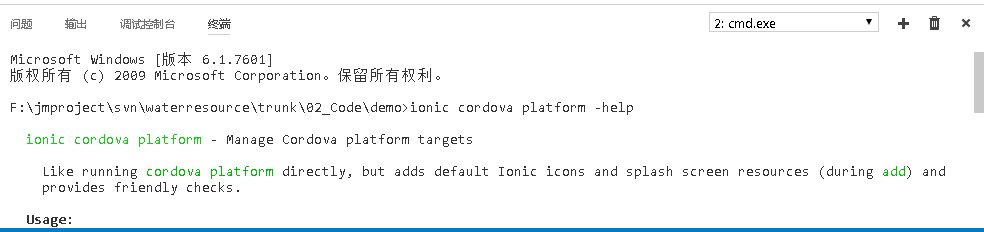
新建的文件一般有4个文件夹，分为html页面，scss样式文件，ts（typescript类似于js

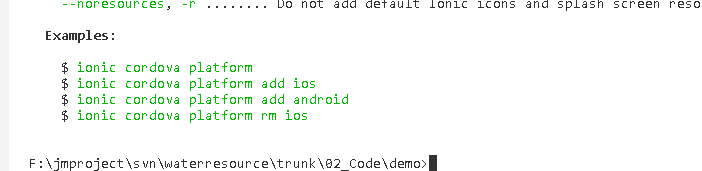
）文件,还有module.ts资源依赖文件。

前端开发api在ionic官网api栏目里面可以查找到相关demo进行学习。

# 平台的添加

在visual studio终端里面运行命令ionic cordova platform -help即可查看所有关于平台操作的命令

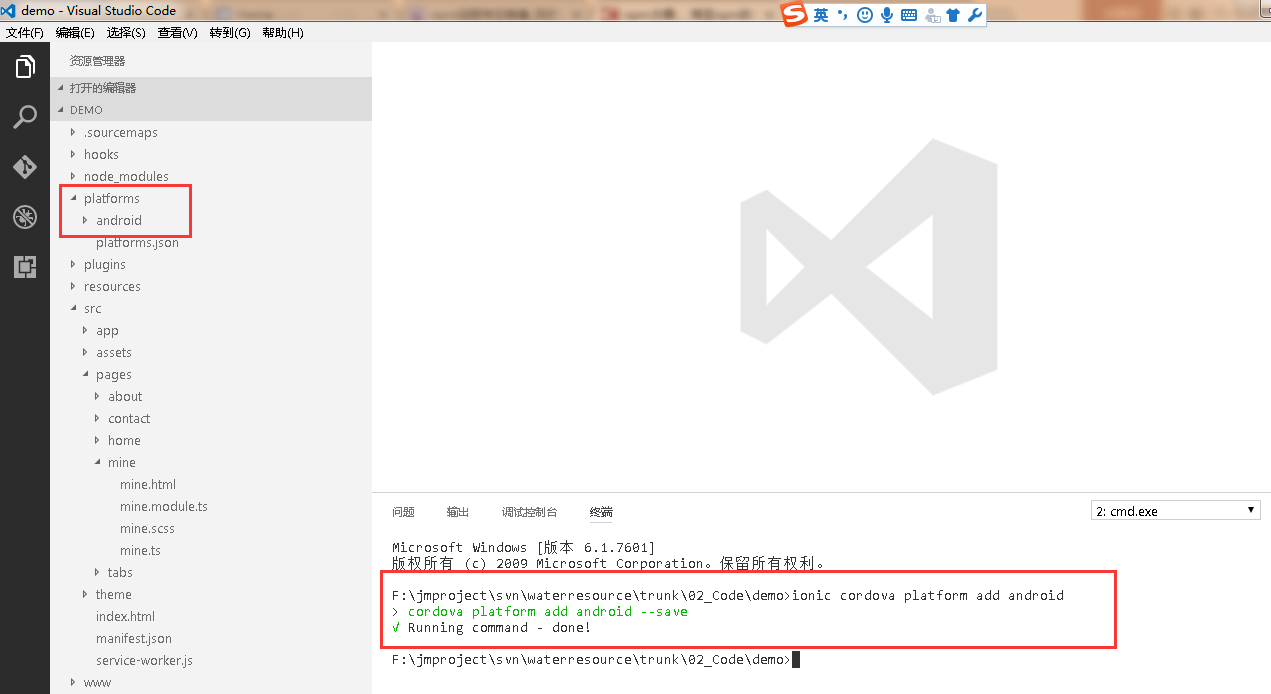




## 1、平台的添加

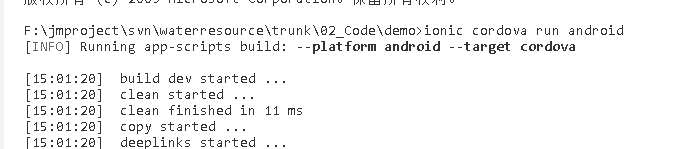
运行 ionic platform add android添加安卓平台 // ionic cordova platform add android

移除：ionic cordova platform rm android



## 2、安装包的创建

安卓平台添加成功后，运行ionic cordova run android





运行完成后 platforms->build->outputs->apk文件夹中的.apk文件即为安卓手机安装包，可以装到安卓手机上进行测试。

## 3、apk签名

### 1、cmd 打开命令行，选择一个路径运行以下命令：

keytool -genkey -v -keystore 你起的名字.keystore -alias 你起的别名 -keyalg RSA -validity 10000

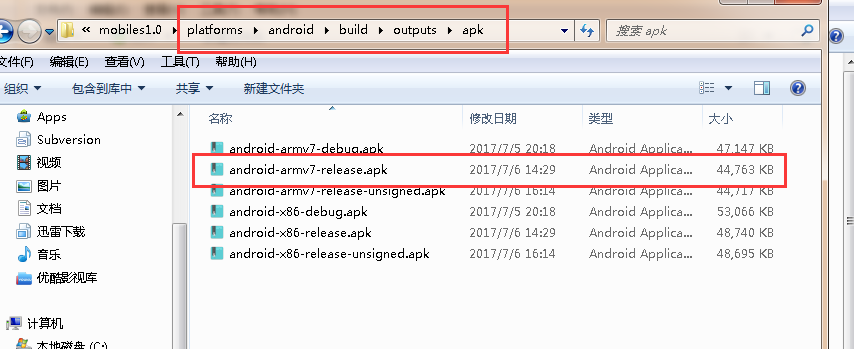


生成一个属于自己独立的签名文件

### 2、在项目中根目录下将刚刚生成的签名文件复制进来，运行以下命令：

打开cmd 到jdk的bin 目录

jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore jiamail.keystore android-armv7-release-unsigned.apk you\_name



# 第三方插件的添加

Ionic官方文档第三方插件地址

[http://ionicframework.com/docs/resources/third-party-libs/](http://ionicframework.com/docs//resources/third-party-libs/)

在GitHub官网搜索插件demo与用法，down下来，添加资源运行demo按照插件介绍进行运用

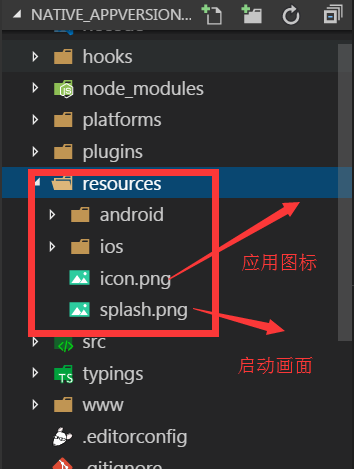
查看某个第三方插件相关的插件都有什么的命令行是：npm search libname

添加native插件：ionic plugin add cordova-plugin-contacts

添加第三方插件：npm install echarts --save

# app应用图标替换

如下图所示，项目目录：



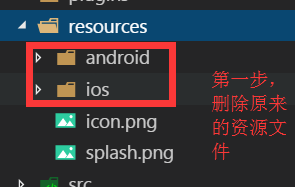
**icon.png:** 为app图标，大小参见原图片格式及尺寸。

**splashpng:** 为启动画面，大小参见原图片格式及尺寸。

**android:** 由icon和splash生成的安卓相关资源文件。

**ios:** 由icon和splash生成的苹果相关资源文件。

**修改app图标或启动画面步骤：**

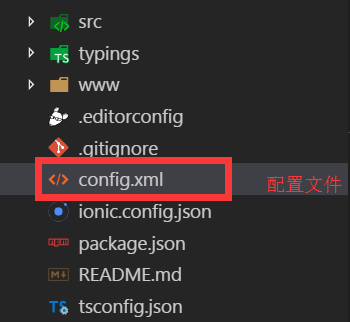
1. 删除android和ios文件夹：
2. 替换icon.png和splash.png(需要换那个就以相同格式文件替换)。
3. 移除对应平台：。  
   android：ionic cordova platform rm android  
   ios：ionic cordova platform rm ios
4. 重新添加平台：目的是重新加载资源文件。

android：ionic cordova platform add android  
ios：ionic cordova platform add ios

1. 打包：执行run方法。

# 版本更新

如下图：在项目的config.xml,及项目配置文件





## 1.<widget >标签中属性:

id=”com.jamail.ionicdemo” ----即应用程序包名-》安装到手机后再手机文件管理中显示的app文件夹名称

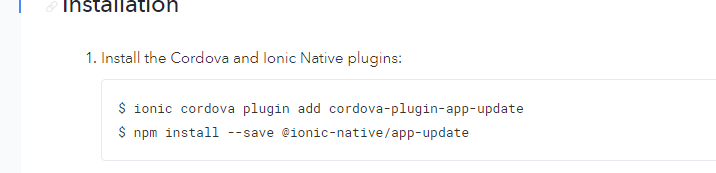
version=”1.0.0” ----应用程序版本号

## 2. <widget >标签中的子标签：

<name>应用程序名称</name>->手机app安装后显示的名字

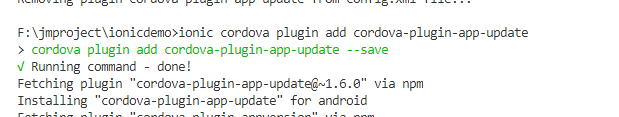
## 3.版本更新：支持android

在ionic官网中https://ionicframework.com/docs/native/app-update/介绍了版本升级的步骤，安装步骤操作即可。



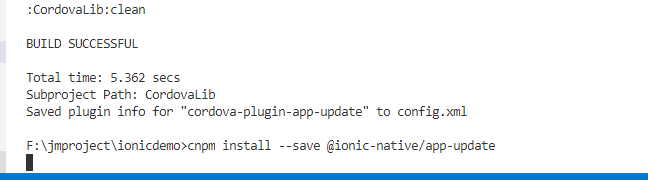
### 用命令安装版本升级插件

ionic cordova plugin add cordova-plugin-app-update



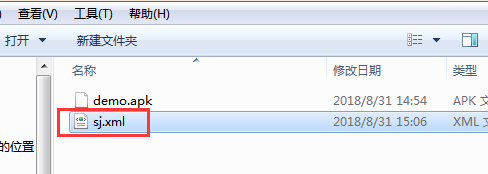
### 用命令更新插件资源

cnpm install --save @ionic-native/app-update

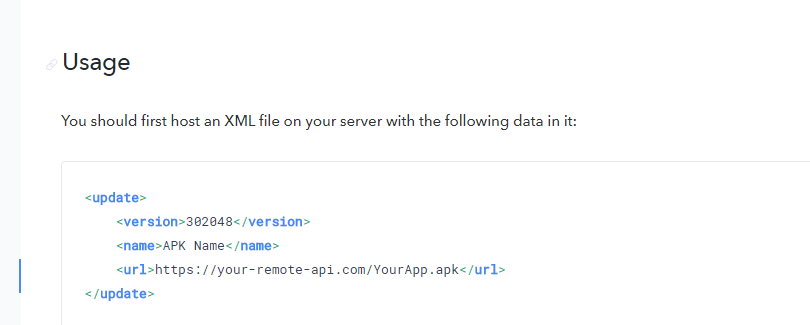


### 制作升级版本文件xml文件

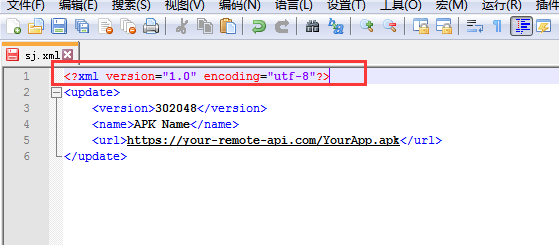
新建文本文档，修改文本文档文件类型后缀为xml



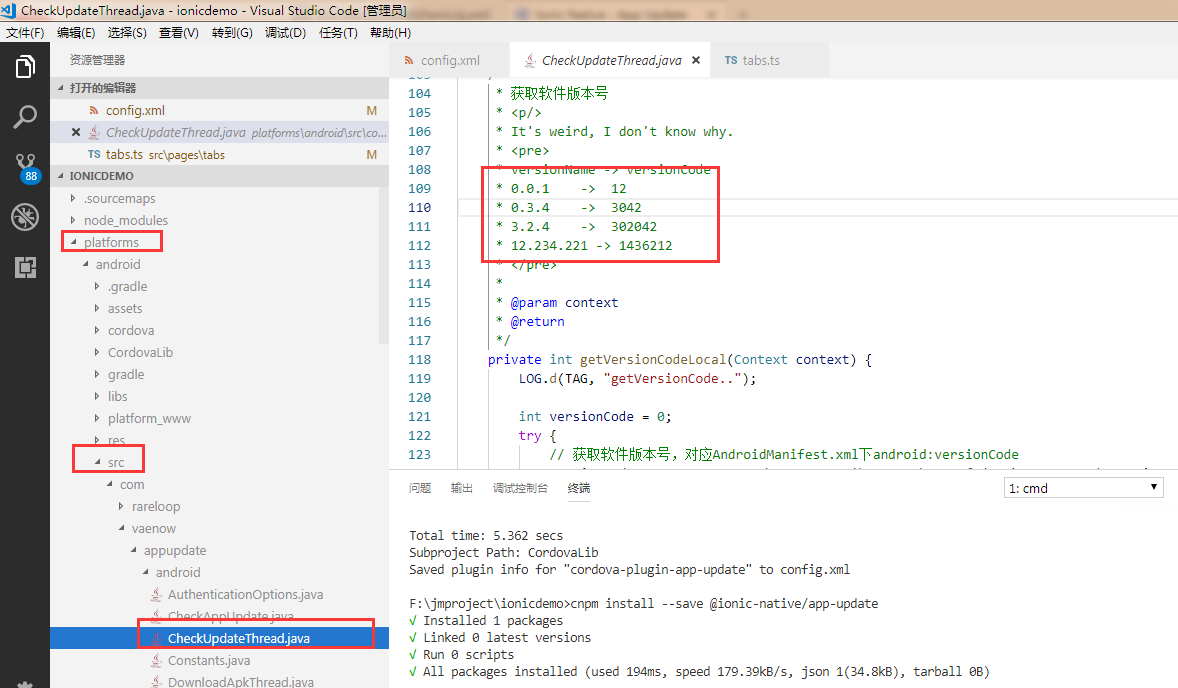
打开xml文件，将官网xml文件代码进行粘贴



其中要补充xml文件代码

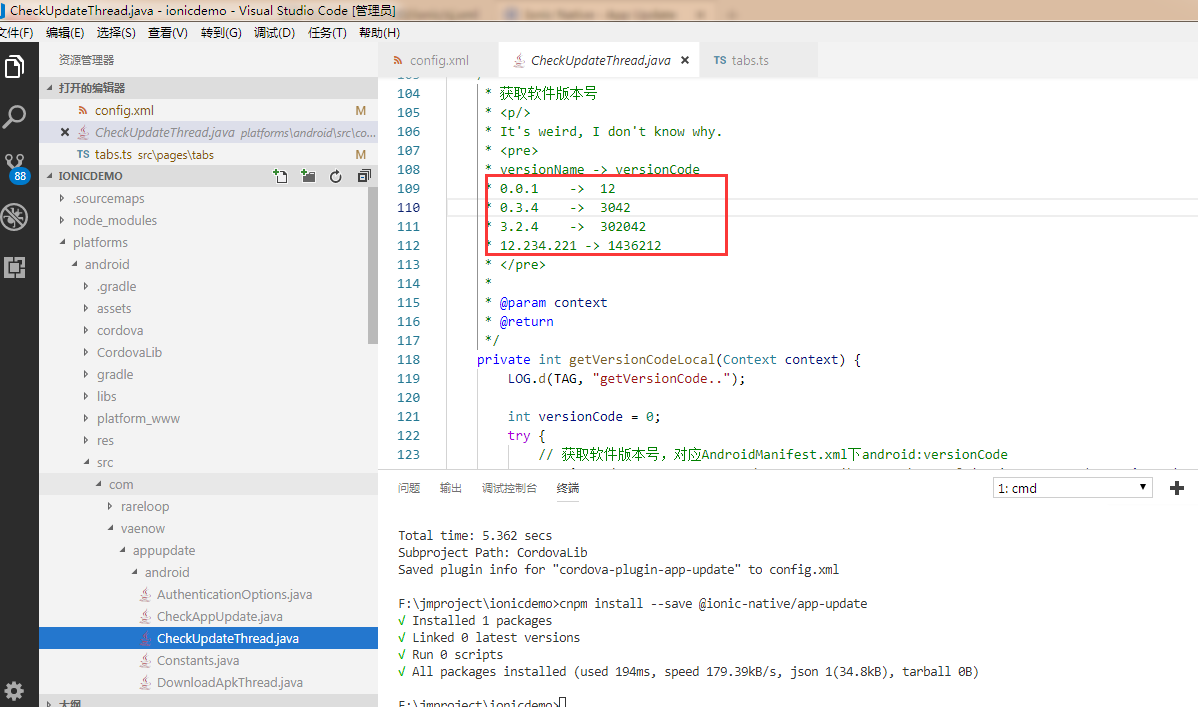


这里的version为当前apk的版本号，版本号的规则按照如下文件进行设置



文件路径为

F:\jmproject\ionicdemo\platforms\android\src\com\vaenow\appupdate\android\CheckUpdateThread.java



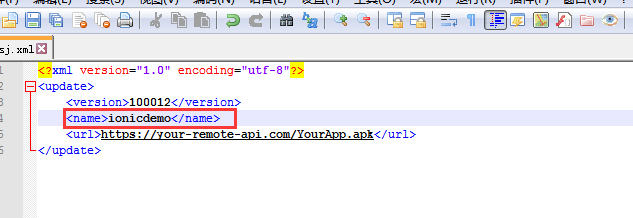
其中版本号的“.”用数字“0”进行替换，前面都为0的话自动省略，最后一位为数字“2”

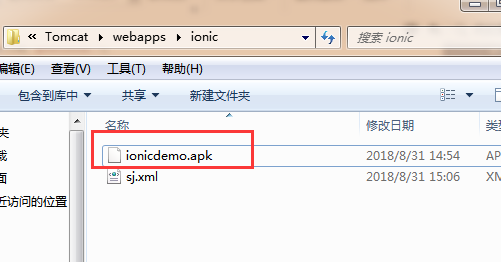
则当前config.xml的版本号：1.0.0按照规则版本号为100002

如果我们对版本号进行升级了，那么服务器上xml中对应的版本号要做相应的修改

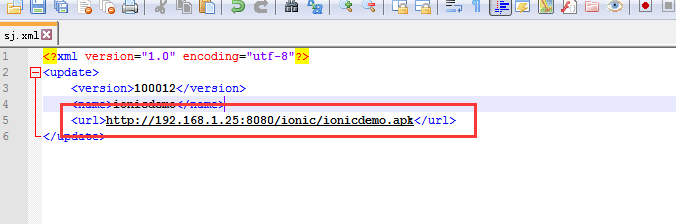
如果将config.xml的版本号升级为1.0.1，则服务器升级的xml文件中的版本号按照规则应为100012。

Name标签中内容为与当前xml文件同一文件夹下的apk文件的名字

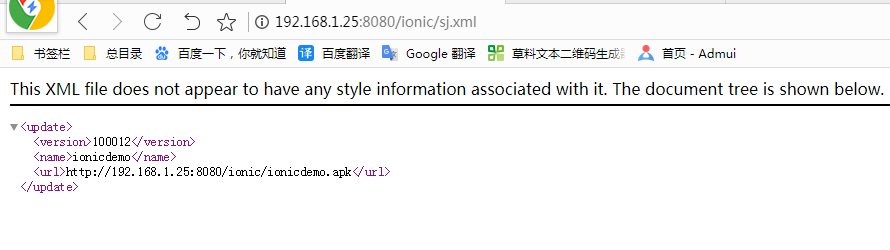




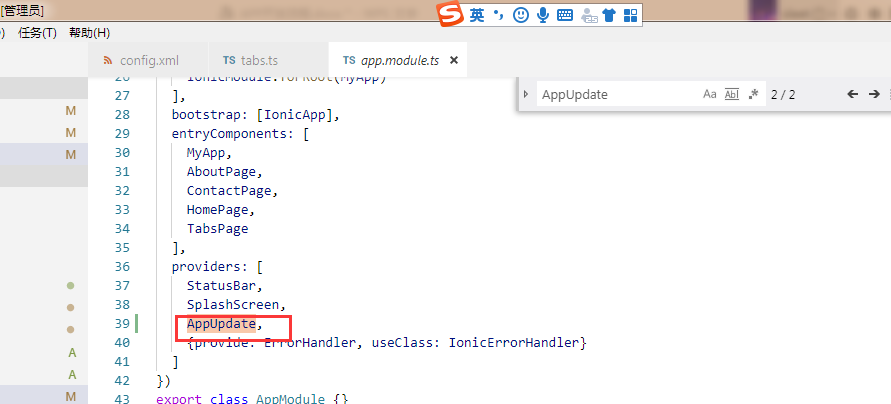
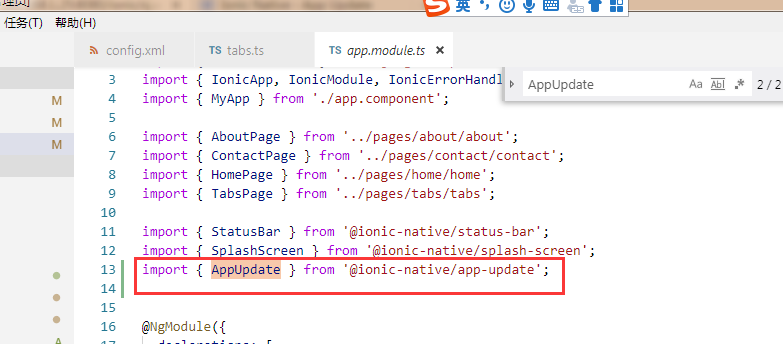
Url为当前同级文件夹下的apk在服务器上的路径



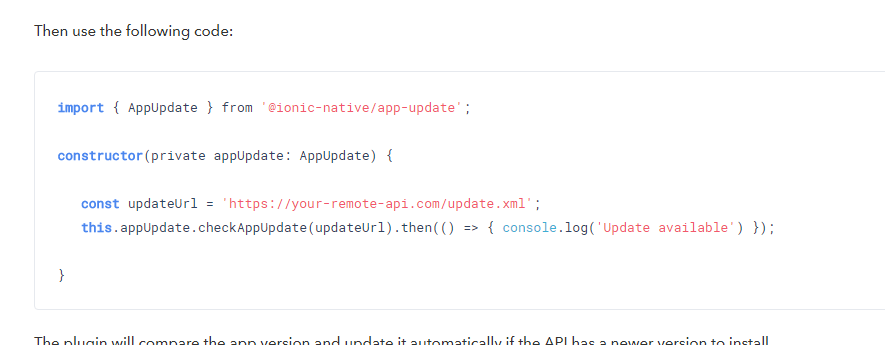
制作好了xml后，可以将文件在浏览器中运行，没有报错及表示文件设置正确

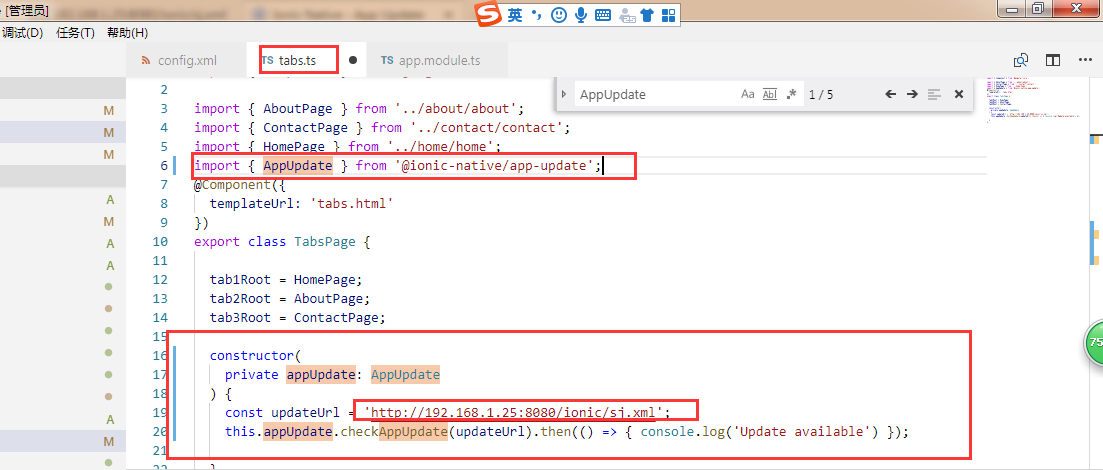


### 在app.module.ts文件中查看升级插件是否引入正确



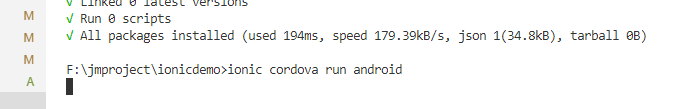
### 在tabs页面中引用插件并按照官网步骤进行代码配置



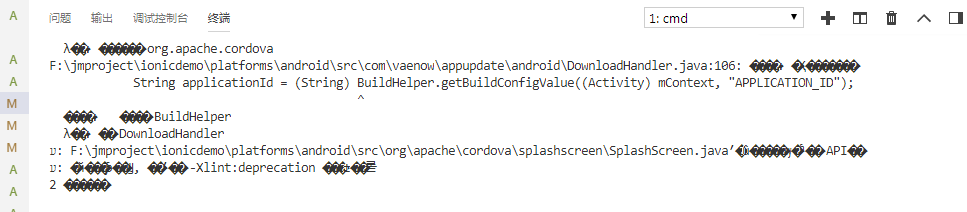


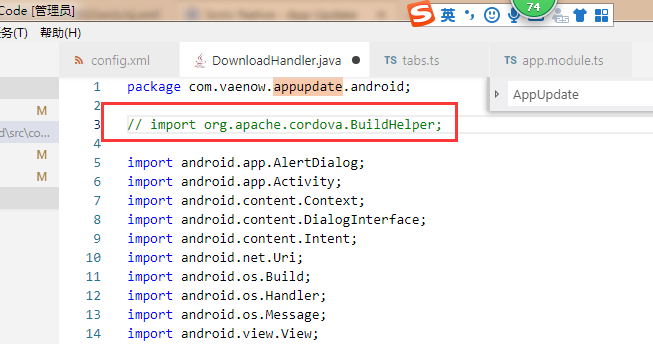
这里要注意将updateUrl路径修改为制作的xml文件的服务器路径。

### 一切操作完毕后运行打包命令进行打包

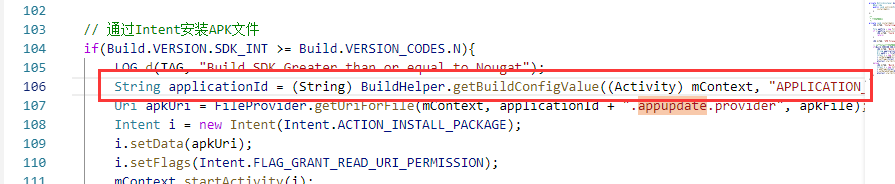


这里有两个文件报错了，报错的是加载的升级插件的错误，我们点击alt键进入文件对代码进行修改

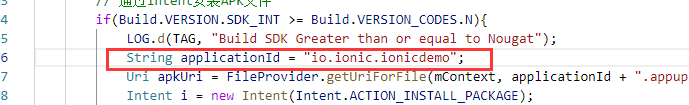




注销此行代码



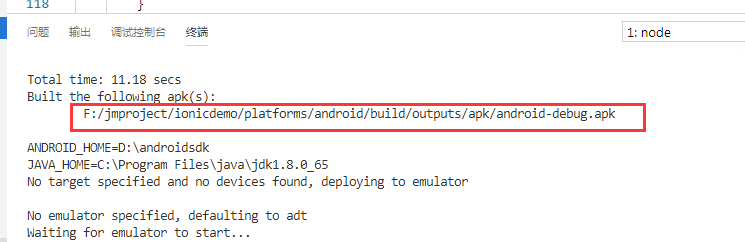
改为自己在config.xml中设置的id的名称，必须为双引号。



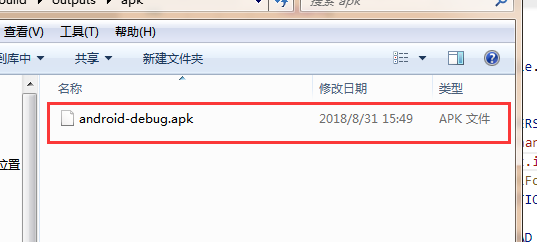
重新运行打包命令

ionic cordova run android / ionic cordova run android --release

打包完成



安装包路径为上图所示



这个安装包就有了监测版本升级的功能，当运行tabs时回去查找xml文件，xml文件中的版本号如果与安装的版本号不一样，则会提示更新。

以后改完代码后，只需要修改config.xml文件中的version，例如我们把当前打包的version为1.0.0的安装包装到了手机上，然后我们打了1.0.1的包，在服务器上修改了xml的版本号代码，并且将1.0.1的包在服务器上进行替换。那么当手机运行1.0.0的软件时会自动提示更新。

# 极光推送

参考文档 <https://blog.csdn.net/m0_37975886/article/details/78970219>

## 在极光推送官网https://www.jiguang.cn/完成注册做极光推送消息发送准备

## 在ionicdemo中终端依次运行以下命令

### ionic cordova plugin add [cordova-plugin-jcore](https://github.com/jpush/cordova-plugin-jcore)@1.1.12

### cordova plugin add [jpush-phonegap-plugin@3.3.2](mailto:jpush-phonegap-plugin@3.3.2) --variable APP\_KEY=your\_jpush\_key

### cnpm install --save @jiguang-ionic/jpush

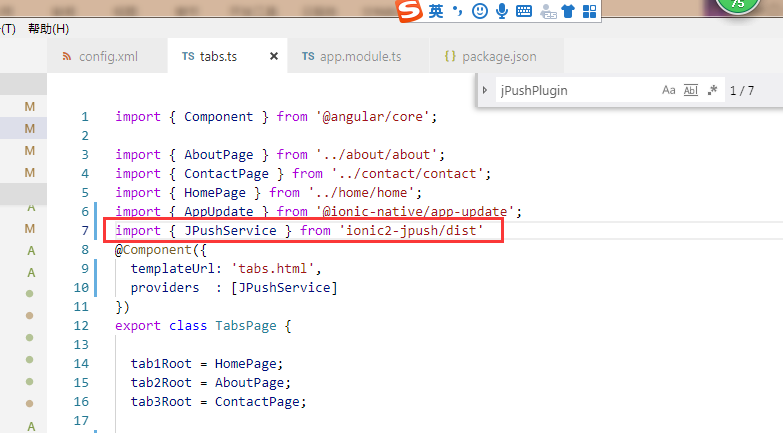
### cnpm install ionic2-jpush --save

因为新的android手机版本与最新的极光推送插件不能适配，所以需要安装指定的插件版本。如果添加的jpush需要卸载则先卸载2，再卸载1。并且卸载了2，如果要重新安装，必须再重新按照步骤装一遍，因为依赖的问题会导致卸载时依赖问津会被卸载掉。

## 配置app.module.ts，注入jpush

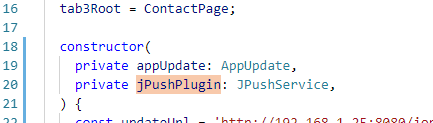
## 4、在tabs中做极光推送调用极光推送初始化方法

### 插件引用import { JPushService } from 'ionic2-jpush/dist'



### 插件注入providers : [JPushService]，private jPushPlugin: JPushService





### 调用初始化极光推送方法

this.jPushPlugin.init()

.then(res => alert(res))

.catch(err => alert(err))



# 公众号网页授权

参考链接https://mp.weixin.qq.com/wiki?t=resource/res\_main&id=mp1421140842

https://open.weixin.qq.com/connect/oauth2/authorize?appid=wx520c15f417810387&redirect\_uri=https%3A%2F%2Fchong.qq.com%2Fphp%2Findex.php%3Fd%3D%26c%3DwxAdapter%26m%3DmobileDeal%26showwxpaytitle%3D1%26vb2ctag%3D4\_2030\_5\_1194\_60&response\_type=code&scope=snsapi\_base&state=123#wechat\_redirect

在请求网页时先调用此接口



点击链接，允许授权后如果用户同意授权，页面将跳转至 redirect\_uri/?code=CODE&state=STATE，并且会在参数中返回code参数和其对应的值，再用code参数发送http请求

获取code后，请求以下链接获取access\_token： https://api.weixin.qq.com/sns/oauth2/access\_token?appid=APPID&secret=SECRET&code=CODE&grant\_type=authorization\_code



