## 开发

### Adb端口被占用

1. 查看5037端口的进程 netstat -ano|findstr “5037”
2. 
3. 打开任务管理器杀掉其他进程即可

### Adb devices 设备offline

浪费大半天的时间，第二天正要重装系统时尝试着看adb版本是不是有问题，果然，还真是adb的，可能是那些刷机软件写入到系统环境下的adb属于自己修改的版本，我android sdk配置的环境变量并没有生效

1. 设备未认证PC调试（没有保存adbkey.pub）
2. Adb问题，尝试adb kill-server
3. 环境问题 重启设备或电脑
4. Adb版本问题（旧或其他修改的版本），尝试正确版本

### 修改手机hosts文件

替换掉/system/etc/hosts即可

1. 获取权限mount -o remount,rw /system
2. 覆盖文件 adb push hosts /system/etc/hosts

## Android studio设置

Studio可以搜索设置的选项，所以知道了关键字，摸索着改下就有了。例如改字体，输入font即可，总不至于在color下乱翻，毕竟也是学过好多年英语的。

### 初始化设置

第一次用studio的话改下这几个参数，让你的studio飞起来

1. 不检查sdk更新
   1. studio安装目录下idea.properties，添加disable.android.first.run=true
2. 开启gradle单独的守护进程

# Project-wide Gradle settings.

org.gradle.daemon=true

org.gradle.jvmargs=-Xmx2048m -XX:MaxPermSize=512m

-XX:+HeapDumpOnOutOfMemoryError -Dfile.encoding=UTF-8

org.gradle.parallel=true

org.gradle.configureondemand=true

* 1. 用户目录下.gradle/gradle.properties

1. 扩大jvm大小
   1. Studio安装目录下studio.exe.vmoptions修改

-Xms256m

-Xmx2048m

### 关联源码

1、关掉你所有的工程，进入AndroidStudio的欢迎界面，依次进入Configure>Project Defaults>Project Structure>SDKs，如果你能看到SDKs选项，基本上选中你想修改的SDK API，找到对应的sourcepath面板就能修改或者增加你的source目录了。

2、如果你和我一样悲剧，在1方法中的Project Structure中看不到SDKs这个选项，那我们就自己动手进入androidstudio的配置目录

windows下：系统盘:\Users\username\.AndroidStudioBeta\config\options\jdk.table.xml

Mac下：~/Library/Preferences/AndroidStudioBeta\config\options\jdk.table.xml

在这个目录下搜索jdk.table.xml，用记事本打开，修改了里面对应的SDK API的sourcePath就行了，点击保存，完事！

### Inspection检查规则，警告 错误

经常看到代码框右侧红色错误或者shi黄色的警告，其实有些只是注释或者native方法，不需要报红色的。所以就需要手动修改检查的规则。

1. 搜索inspection，找到java issues--declaration has problem in java doc reference 就是方法注释参数的问题。
2. 搜索corresponding，看到android--missing jni function 就是native方法找不到的问题

还有其他非常多的规则，如果不知道关键字，真的是跟瞎猫捉耗子一样，所以还是要好好学习english

### Sanning file to index

这个问题应该是bug导致的无响应，FIle--Invalidate Cache清除缓存即可

### 常规设置

1. **代码提示不区分大小写**：code completion
2. **编码**：encoding，一共四个地方：IDE Project Directory Property 另外studio偶尔也会出bug，尽量不要同时打开编码不同的项目
3. **导出studio设置**：studio重装了，原来所有的配置都还在，真是美呆了 FIle--import setting
4. **书签**：添加删除ctrl+shift+数字键 查看ctrl+数字键
5. **类型转换**：alt+enter
6. **导包**：ctrl+alt+O ctrl+shitf+O
7. **方法注释**：fix doc commnet
8. **打包可调试版本**：在module的build.gradle中release节点下修改minifyEnabled false

debuggable = true即可。当然这个版本肯定是没有混淆的，不然还调试个鬼啊

### 常见错误

1. Execution failed for task ':app:dexDebug'.Process 'command 'C:\jdk1.8\bin\java.exe'' finished with non-zero exit value 2
   1. 乍看是jdk的问题，其实不然，引起这个的问题有很多，比如jvm内存溢出，引用了重复的jar包，工程dex方法数超过65536等。可以在app module下的builde.gradle文件中android节点下加入dexOptions{javaMaxHeapSize ”2048M”},这个不用说也知道是防止内存溢出的。在defaultConfig中加入mutilDexEnable true,防止方法数超过而编不过
2. Execution failed for task ':app:dexDebug'.Process 'command 'C:\jdk1.8\bin\java.exe'' finished with non-zero exit value 1
   1. 跟上面的问题很相似，但是退出值是1，网上给的解决方案也是各说各的
      1. The right answer is, that some of your jar files does not compile. You should go into your build.gradle file in your project, and look in your dependencies.If you're just importing some jar files, you could try to remove them and add them one at a time. This will help you determine which one of them causes the error.这位老兄的意思是引用的jar编译不过。用最笨的方法一个个去掉再加上，看是哪个jar的问题。另外也有可能是jar重复了
      2. modify the jdk version of your project，另外可能是jdk的问题，在app的build.gradle中buildType的同级节点添加或修改jdk的版本，compileOptions {sourceCompatibility JavaVersion.VERSION\_1\_7 targetCompatibility JavaVersion.VERSION\_1\_7}

## 其他

### Word红色波浪线

选项--拼写检查