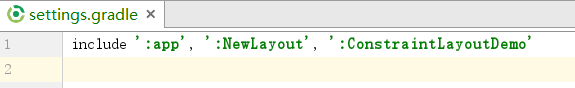
# 第一章 Android开发 Gradle基础

## 1.1 Android中的Gradle构建文件

创建一个新的Android项目时，会生成3个Gradle文件：settings.gradle、build.gradle、app/build.gradle

### settings.gradle

配置了项目中包含的模块，如下：



### 顶层的build.gradle

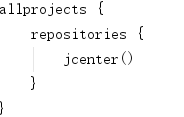


用于为所有的模块设置通用的配置

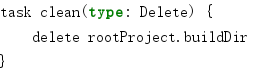


buildscript翻译过来为“构建脚本”，这个代码块就是用于配置从哪下载构建脚本插件，从图可知是从jcenter仓库中下载，下载的是gradle插件，这个gradle插件就是“构建脚本”，用于构建Android项目。

jcenter即Bintray Jcenter Artifactory库。也支持其他的库（特别是mavenCentral()，默认的Maven仓库），但是Jcenter现在已经成默认的了。Jcenter中的内容都使用HTTPS连接并通过CDN来提供，同时也更快速。



allprojects配置了所有模块都默认使用jcenter仓库来下载Java依赖库



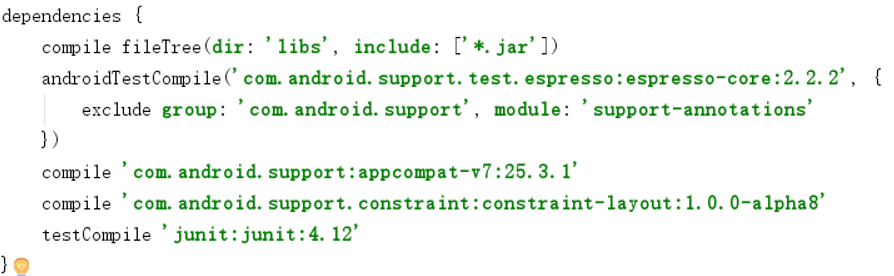
Gradle允许自定义task，上面代码就是一个自定义的task，type: Delete表示这个task是Gradle内置的Delete task的一个实例，在这个task 中，它会把项目根目录下的build目录删除。

### 模块里的build.gradle





通过Gradle的apply功能，将Android插件添加到构建系统中，从而能够使用Android的领域特定语言（Domain Specific Language, DSL）配置。



dependencies用于配置依赖，compile fileTree用于依赖一整个目录树里的文件，dir指定了依赖的目录，include为过滤器，表示只依赖\*.jar文件。

testComile 用于指定测试的依赖，也就是说这些依赖只能在src/AndroidTest/java或者src/test/java，这意味着打包apk时这个测试的依赖jar不会打包到apk中

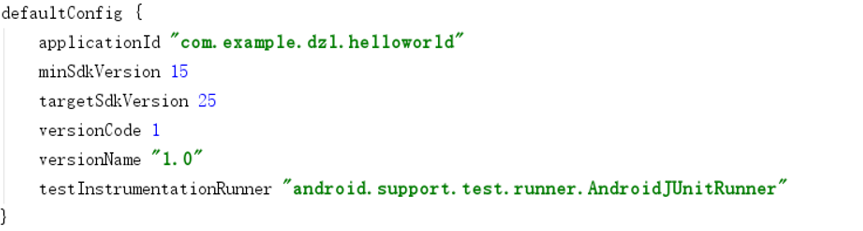
## 1.2 配置SDK版本以及其他的默认值

顶层的build.gradle通过buildscript块将Gradle的Android插件添加到了项目中，模块中的build.gradle通过apply应用了这个插件，因此可以将android块添加到Gradle的DSL中。在android块里面，你可以指定几个项目属性，如下：



android块是Android DSL的入口,compileSdkVersion和buildToolVersion指定编译SDK版本和构建工具版本，这些版本应该指定为最新的版本，因为它们是身后兼容的，并且包含了之前版本的漏洞修复。

defaultConfig块：



applicationId 应用的唯一标识。应用一但上线，这个id就不能修改。一修改就相当于是另一个应用了。

minSdkVersion 应用程序支持的最小Android SDK版本，小于这个版本的设备在访问Google Play商店时将不会看到这个应用程序。

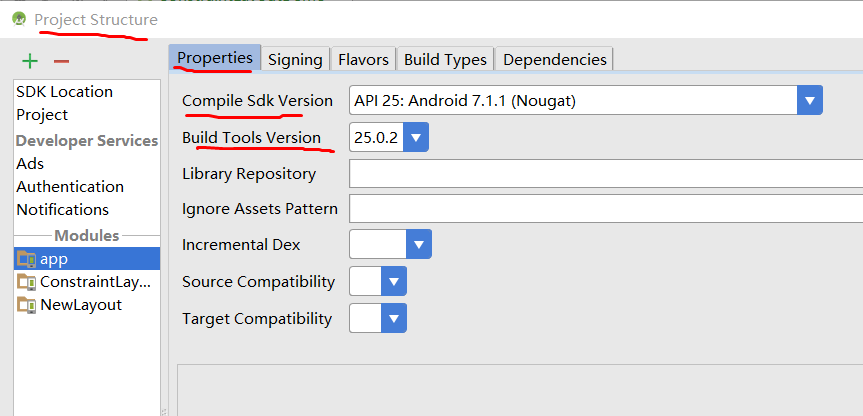
targetSdkVersion 应用程序的目标版本号。如果没有指定为最新版本，AS会给出警告，但是你可以作用任何的版本。

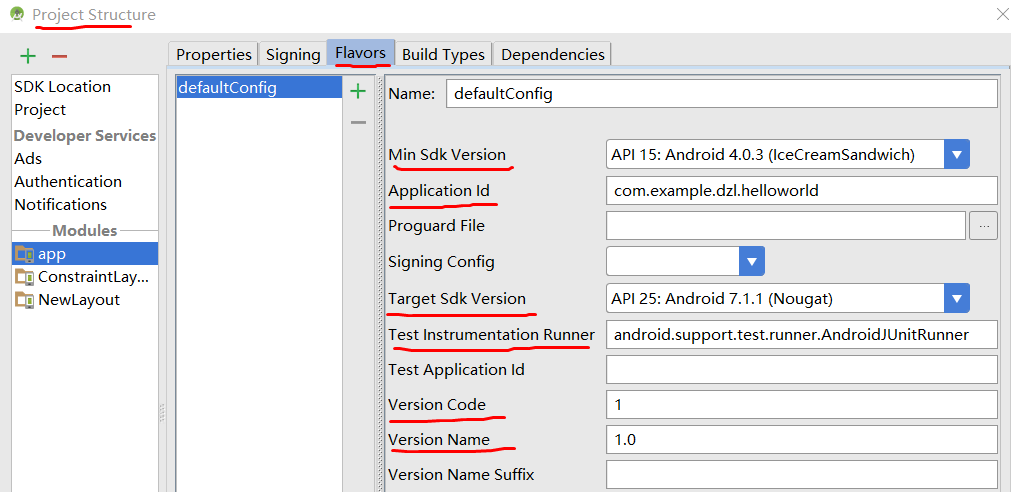
versionCode 应用程序的版本号，用于升级

versionName 版本号的名字，用于显示给用户看，通常的格式为：<major>.<minor>.<version>，就像大多数的项目一样。

testInstrumentationRunner

这些属性在AS的项目结构窗口中有图形化的显示，如下：





defaultConfig块的文档，以及其他元素的DSL，都可以在DSL参考中找到：<http://bit.ly/gradle-dsl> 或 <http://google.github.io/android-gradle-dsl/current/index.html>

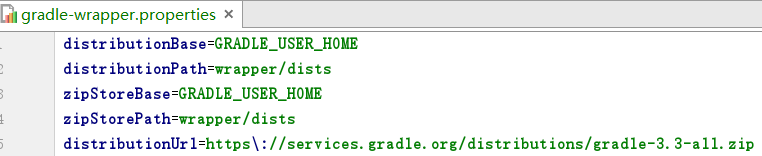
## 1.3 在命令行执行Gradle构建

在命令行执行Gradle构建，使用的是AS提供的Gradle wrapper或者安装Gradle直接运行。你不需要为了构建Android项目而安装Gradle，AS包含了一个Gradle的发行版本（以一个插件的形式）并包含了专门的功能来支持它。

“Gradle wrapper”引用gradlew脚本和gradlew.bat脚本，bradlew给UNIX使用，gradlew.bat给Windows使用，这两个项目在项目的根目录中，结尾的“w”代表“wrapper”。

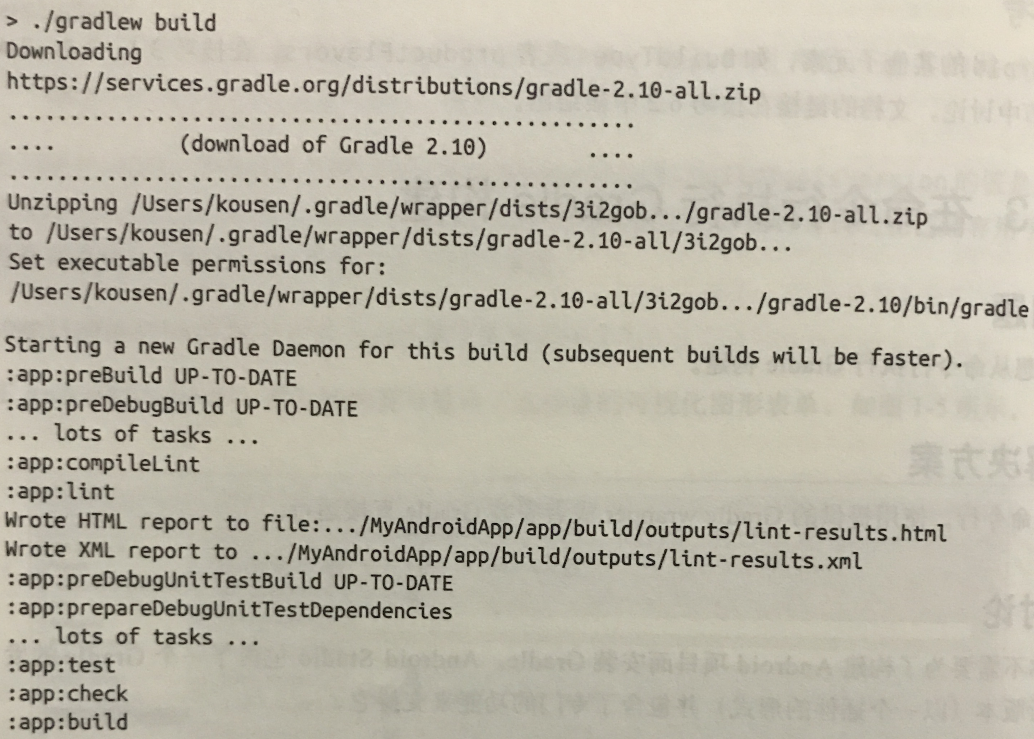
Gradle Wrapper的第一个目的就是允许客户端无须安装Gradle即可运行Gradle脚本（其实所谓的无需安装只是说不需要我们手动去安装，AS会根据wrapper中的配置去下载gradle并安装）。wrapper使用了项目根目录下的gradle/wrapper下的gradle-wrapper.jar和gradle.wrapper.properties文件来启动进程。

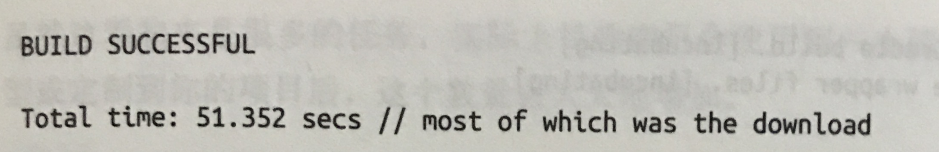
gradle.wrapper.properties文件内容如下：



distributionUrl显示了wrapper会下载并安装Gradle 3.3(可以修改为任意可用的版本),下载保存到zipStoreBase下的zipStorePath目录中，这样3.3版本只需要下载一次，以后如果是相同的版本就不需要再下载了。变量GRADLE\_USER\_HOME即电脑上的用户目录，如：C:\Users\EvenDai

假设把上面的版本修改为2.10，然后在命令行执行gradlew build，输出结果如下：





从上面截图可知，gradlew build是执行了一个脚本命令，并不是Gradle任务哦，这个脚本命令首先要下载gradle 2.10，然后解压，然后开启了一个Gradle的后台进程，然后开始执行Gradle tasks（只有开启了Gradle进程，才能执行Gradle tasks）。

可以在命令行运行任何支持的Gradle tasks，包括你自定义的tasks。编译的代码在app/build目录下，生成的apk(Android package)在app/build/apk目录下。