

Gradle系列之Android Gradle插件



躬行之 发布于 2020-08-23

原文发于微信公众号 jzman-blog，欢迎关注交流。

通过前面几篇文章学习了 Gradle 基础知识以及 Gradle 插件相关的知识，关于 Gradle 及其插件相关知识请先阅读下面几篇文章：

- [Gradle系列之初识Gradle](#)
- [Gradle之Groovy基础篇](#)
- [Gradle系列之构建脚本基础](#)
- [Gradle系列之认识Gradle任务](#)
- [Gradle系列之Gradle插件](#)
- [Gradle系列之Java Gradle插件](#)

学习 Gradle 的目的主要是为了更好的在 Android 开发中使用 Gradle，这篇文章的主要内容是 Android Gradle 插件的相关知识，主要内容如下：

1. 了解 Android Gradle 插件
2. Android Gradle 插件分类
3. 使用 Android Gradle 插件
4. Android Gradle 工程目录
5. Android Gradle 基本配置
6. Android Gradle 任务

了解 Android Gradle 插件

顾名思义，Android Gradle 插件就是一个用于 Android 构建项目的一个 Gradle 插件，该插件有 Google Android 开发团队开发，而且 Android 开发 IDE Android Studio 就采用 Gradle 构建项目，下面是 Android Gradle 插件的优势：

1. 方便重用代码和资源；
2. 可更方便的创建应用的衍生版本，如多渠道打包；
3. 配置方便。可以扩展，还可以自定义构建过程
4. Android Studio 与 Gradle 深度融合

Android Gradle 插件分类

Android 插件的分类是根据 Android 工程的属性进行分类的，Android 工程分为三类，具体如下：

1. App 应用工程：可生成可运行的 apk ；
2. Library 库工程：可生成 aar 共其他 App 应用工程使用，使用方式和 jar 一样，里面有相关的 Android 资源文件
3. Test 测试工程：用于 App 应用过程或 Library 工程进行测试。

对应的就有三种不同的 Android Gradle 插件，三类插件分别是 App 插件、Library 插件和 Test 插件，其插件 id 如下：

```
//App插件id
com.android.application
//Library插件id
com.android.library
//Test插件
com.android.test
```

使用最多的肯定就是 App 插件和 Library 插件，使用这几个插件在 Android Studio 中就可以构建 Android 工程了。

使用 Android Gradle 插件

Gradle 插件使用时以插件的 id 作为唯一标识，如果是第三方的插件，还必须在 buildscript{} 中指定其 classpath，Android Gradle 插件就属于第三方插件，使用时要指定其 classpath，Android Gradle 插件托管到 jcenter 上，所以使用时要指定对应仓库，具体参考如下：

```
buildscript {
    //配置插件仓库
    repositories {
        google()
        jcenter()
    }
    //配置依赖
    dependencies {
        classpath 'com.android.tools.build:gradle:3.0.1'
    }
}
```

配置好第三方插件的仓库和依赖，然后使用 apply 使用该插件，具体如下：

```
//使用App插件
apply plugin: 'com.android.application'

android {
```

```
//编译所依赖的Android SDK的版本
compileSdkVersion 26
//构建工具的版本
buildToolsVersion '26.0.2'
//...
}

dependencies {
    //...
}
```

在 Android 开发中一般将插件的仓库和依赖配置到根项目下的 build.gradle 文件中，其子 Module 项目将不需要在配置了，可直接使用，这里仅仅是 App 插件的使用，当然 Libray 插件的使用和 Test 插件的使用也是一样，使用时指定对应的 plugin id 即可。

Android Gradle 工程目录

下面是 Android Studio 创建 Android 工程的 Module 的目录，这也是 Android 工程的基本目录结构，参考如下：

```
app
├─libs
├─proguard-rules.pro
├─build.gradle
└─src
    ├─androidTest
    │   └─java
    ├─main
    │   ├─AndroidManifest.xml
    │   ├─java
    │   └─res
    └─test
        └─java
```

上述目录中 src 下面的 main、androidTest、test 是三个 SourceSet，main 表示 App 的源代码及资源目录，android 和 test 表示 Android 相关测试代码目录，此外 proguard-rules.pro 是当前项目的混淆文件，libs 用于存放 jar、aar 等库文件，build.gradle 文件用于当前配置文件。

Android Gradle 基本配置

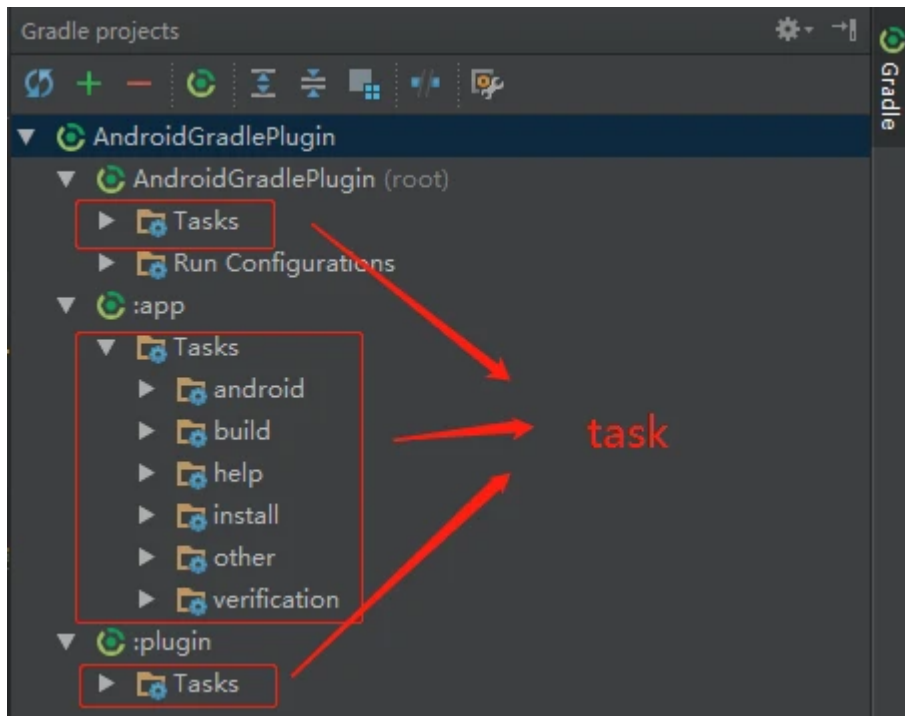
为了介绍 Android Gradle 插件基本配置，使用 Android Studio 创建一个 Android 项目，下面的 app 的一个配置文件，具体如下：

```
//使用Android Gradle插件
apply plugin: 'com.android.application'
//android{}是Android工程配置的入口
android {
    /**
     * 指定编译依赖的Android SDK的版本,同
     * compileSdkVersion android-26
     * android.compileSdkVersion = 26
     * android.compileSdkVersion = 'android-26'
     */
    compileSdkVersion 26
    //指定构建工具的版本,也可以使用属性值buildToolsVersion设置其版本
    buildToolsVersion '26.0.2'
    /**
     * 默认配置, defaultConfig是一个ProductFlavor,可根据不同需求生成不同的Apk
     * 如果不自定义ProductFlavor进行单独配置, 则该productFlavor会采用默认配置生成Apk
     * 这些具体配置, 如applicationID都是ProductFlavor的属性
     */
    defaultConfig {
        //配置唯一包名
        applicationId "com.manu.androidgradleplugin"
        //最低支持的Android系统版本
        minSdkVersion 19
        //配置应用的目标Android系统版本
        targetSdkVersion 26
        //用于控制应用升级的版本号
    }
}
```

上述配置文件中已经标注了一些 Android Gradle 插件需要配置的地方，也算是对 Android Gradle 插件及项目配置文件 build.gradle 的一次理解与学习，关于 Android Gradle 插件的基本配置就到此为止，以后陆续补充。

Android Gradle 任务

Android Gradle 插件基于 Android Gradle 插件，Android Gradle 插件中包含了 Java Gradle 插件的一些任务，如 assemble、check、build 等任务，此外，Android Gradle 插件还添加一些新任务，如 connectedCheck、deviceCheck、lint、install、uninstall 等 Android 开发特有的任务，可在 Android Studio 上查看 Android 开发中一些任务，选中 Android Studio 右侧标签 Gradle 就可查看，参考如下：



这篇算是初步从学习 Gradle 的角度了解了 Android Gradle 插件相关知识，后面的文章中会继续 Android Gradle 的学习。可以关注公众号：躬行之（jzman-blog），一起交流学习。

gradle groovy  android 插件 task

阅读 1.9k · 发布于 2020-08-23