## 安卓Training文档翻译

## 写在前面的话

在当前的互联网浪潮下，一个典型的Chinese Coder似乎多了很多或自黑或被自黑的特质，自觉或不自觉地成了屌丝，技术宅的代名词。作为一个不称职的程序猿，总感觉需要做点什么事情来衬托一下自己的职业。之前的一家公司有公用vpn，看安卓文档似乎顺理成章，几个月前到了一家新公司以后发现没有了这一待遇，于是乎在被百度上面的技术资料折腾了几个月之后忍不住自费开始了翻墙之旅。而作为一个纯屌丝，总是觉得需要对得起这一个月十几块钱的支出，于是乎想到做这么一件事情: 把安卓的官方Training和Design的文档翻译出来，一来强化一下自己对安卓的理解，二来也希望或多或少可以帮助到国内在做安卓开发的同仁们。

对于初学者来说，这两篇文档可以提供非常直观的指引，我相信几乎100%的安卓开发者都是参考着它们进入安卓的世界的。而对于希望进阶的人们来说，这两篇文档很清楚地指出了安卓的设计者希望开发者来使用sdk进行开发的方式，并且透露出一定的安卓设计思想。与其在安卓底层挣扎，不如先把上层的思想吃透，由浅到深，总是“浅”在前，你说呢。

## 让我们开始吧

欢迎来到“安卓开发者训练营”，在这里，你能够找到许多包含示例代码的安卓开发课程。这些课程被分为若干组。

第一组课程，“开始”，可以教会你关于安卓开发的基础内容。如果你倾向于视频教程，请点击：<https://www.udacity.com/course/ud853>

### 你的第一个安卓应用

在你装好安卓SDK后，通过此组课程来学习安卓开发的基础知识。

新建一个安卓工程含有包括程序代码在内的所有文件。这一课程向你展示两种创建安卓工程的方法：通过Android Studio创建或者通过SDK工具在命令行里创建。

再次之前，你需要安装好安卓SDK。如果你希望通过Android Studio来进行创建，你需要安装好Android Studio：<http://developer.android.com/sdk/installing/studio.html> 如果你未曾安装，参考这篇指引：<http://developer.android.com/sdk/installing/index.html>

#### 通过Android Studio创建安卓工程

##### 1、第一步

如果当前Android Studio没有打开任何工程，在欢迎界面，点击新建

如果当前有工程被打开，在文件菜单，选择新建

此时新工程界面出现

##### 2、填好所有信息，点击“下一步”

如果你采用推荐方式填写信息，这一步会方便很多。

Application Name是给用户看的程序名称。在这个程序中，可以采用“我的第一个App”。

Company domain给包名提供了一个前缀。Android Studio会记住你所使用的该前缀。

Package name是该安卓工程的全名，和Java程序语言中的包名一致。你的程序所使用的包名必须在安卓设备上唯一。你可以通过修改程序名或者公司名来修改此包名。

Project location是你用来存放该工程文件的目录。

##### 3、在“选择您的程序运行的设备”选择是手机或者平板

##### 4、在最低版本SDK，选择API8，即安卓2.2（Froyo）

“最低版本SDK”是您的程序所支持的最低版本安卓，代表所使用的API版本。如果你想要支持尽可能多的设备，你应该在能满足你的程序的核心功能的情况下尽可能将其设到最低。如果你的程序有些特性只有在高版本的安卓系统能够支持，并且这些特性不是程序的核心功能，你可以在高版本的安卓手机上开启这些特性。（关于这一点，参考<http://developer.android.com/training/basics/supporting-devices/platforms.html>）

##### 5、暂时忽略其他选项（电视，穿戴，眼镜），点击下一步

##### 6、在“选择Activity模板”，选择空白Activity，点击下一步

##### 7、在“详细信息”里面，将Activity名字改为MyActivity，布局名字改为activity\_my，标题改为MyActivity，菜单资源名称改为menu\_my。

##### 8、点击“结束”按键完成安卓工程的创建。

##### 此时，你的程序是一个基本的“Hello World”程序，包含一些基本的默认文件。下面我们来浏览一下一些比较重要的：

##### app/src/main/res/layout/activity\_my.xml

这个XML layout文件对应着你在创建安卓工程时添加的activity.经过“新建安卓工程”的流程，Android Studio自动创建该文件，并为该文件提供一个内容界面和对应的UI预览。此文件包含一些来自“材料设计库”的默认UI元素，包含“app bar”和一个悬浮的按键。除此以外，它还包含一个单独的含有主要内容的layout文件。

##### app/src/main/res/layout/content\_my.xml

这个XML layout文件被包含在activity\_my.xml内，含有一些设置信息和一个展示“Hello World”信息的TextView控件。

##### app/src/main/java/com.mycompany.myfirstapp/MyActivity.java

在新建项目的工作完成后，该文件会出现在Android Studio的界面上。当你点开它之后，你会发现你所创建的类。当你运行该app的时候，Activity类会启动这个activity并加载含有“Hello World”内容的layout。

##### app/src/main/AndroidManifest.xml

此文件包含app的基础信息和对app内各组件的描述。随着课程的继续，你会在app里面增加更多的组件，此时你会重新查看、修改此文件。

##### app/build.gradle

Android Studio使用Gradle来编译你的app。在你的安卓工程中，每一个模块下面都有一个build.gradle文件，同时还有一个对应于整个工程的build.gradle文件。通常情况下，你只会对整个app相关的build.gradle文件感兴趣，你的app的编译依赖和默认设定通常都写在这个文件里面。常见的设定有：

compileSdkVersion：你的app编译时对应的安卓平台版本。通常情况下，这个值设定为你的当前最高版本的sdk。需要注意的是，这一版本需要大于安卓4.1，如果你没有如此高版本的sdk的话，你需要通过SDK Manager来安装一个。当然，你也可以设定为较老版本的sdk，不过只有将其设为最新版本的sdk你才能用到一些最新的特性，从而让用户得到最好的用户体验。

applicationId：这是你在新建项目过程中为你的app设定的唯一的完整包名。

miniSdkVersion：这是你在新建项目过程中设定的最低支持的sdk版本。即你的app所支持的最低安卓SDK版本。

targetVersion：代表你所测试过的你的app所支持的安卓版本。在新版本的安卓出现的时候，你应该在新版本上测试你的app，然后将此值更新为最新的API版本，从而使用最新的平台特性。关于这方面的内容，参见：<http://developer.android.com/training/basics/supporting-devices/platforms.html>

关于使用Gradle编译的更多信息，参见

<http://developer.android.com/sdk/installing/studio-build.html>  
让我们再来关注一下/res目录的子目录，它们存放有你的app所使用的资源文件。

drawable-<density>/：

存放“可绘制文件”的目录，除了icons，可以适应不同的像素密度

layout/：

存放用来定义你的app的用户界面的文件，如activity\_my.xml。

menu/：

用来存放app的目录描述文件。

mipmap/：

用来存放icon文件，包含默认icon ic\_launcher.png。

values/：

用来存放其他的xml文件，这些xml文件通常含有一系列的资源，包括字符串和颜色定义等。