

1.2 Android Studio 安装与配置

Android Studio 下载地址如下：

官网下载：<https://developer.android.google.cn/studio#downloads>。

下载完之后，双击可执行文件即可进行安装。

(1) 进入后按照向导指示安装，首先出现图 1-12 的欢迎界面。

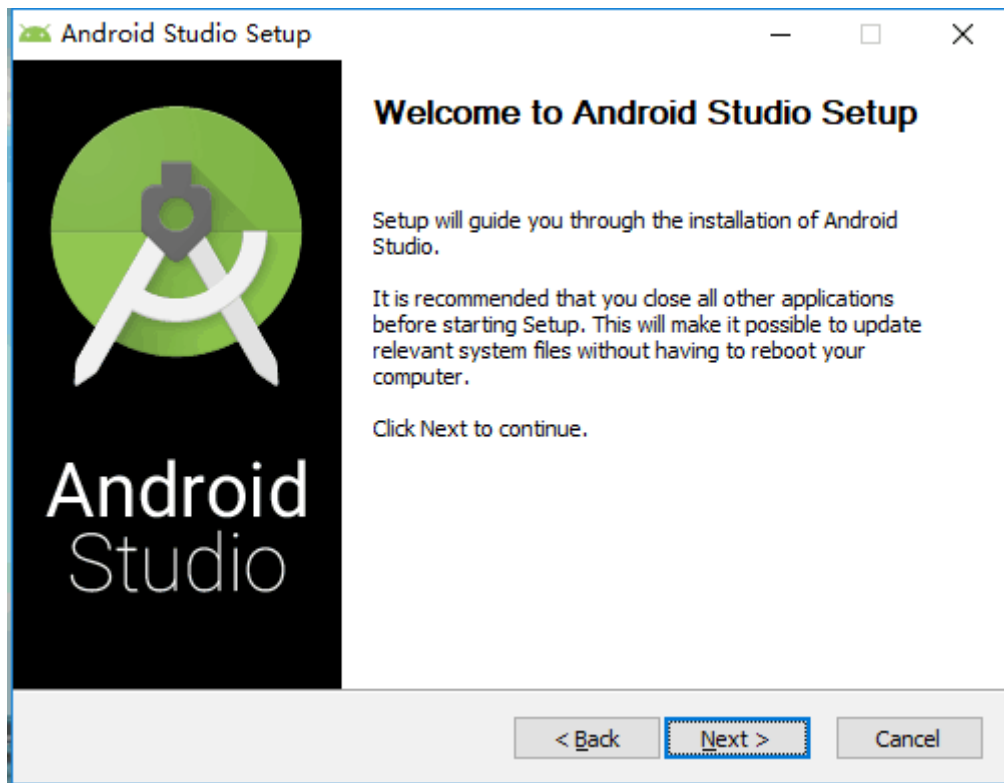


图 1-12 欢迎界面步骤阅读

(2) 进入图 1-13 界面，第一个是 Android Studio 主程序，必选。第二个是 Android 虚拟机，若要安装，也勾上。完成后点击 next 下一步进入图 1-14 界面。

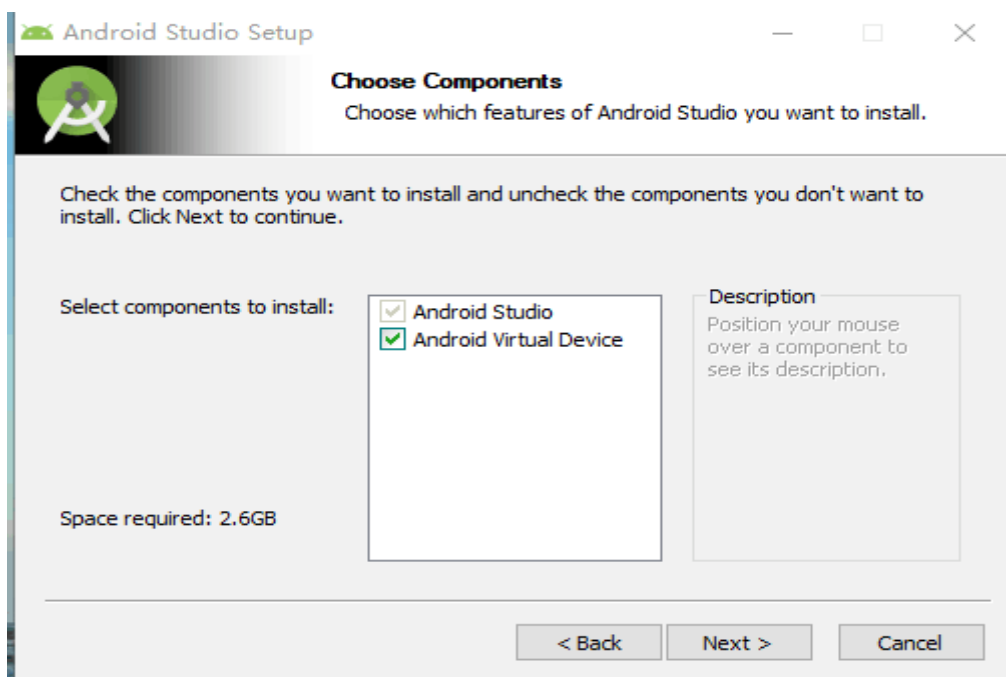


图 1-13 选择需要安装的选项

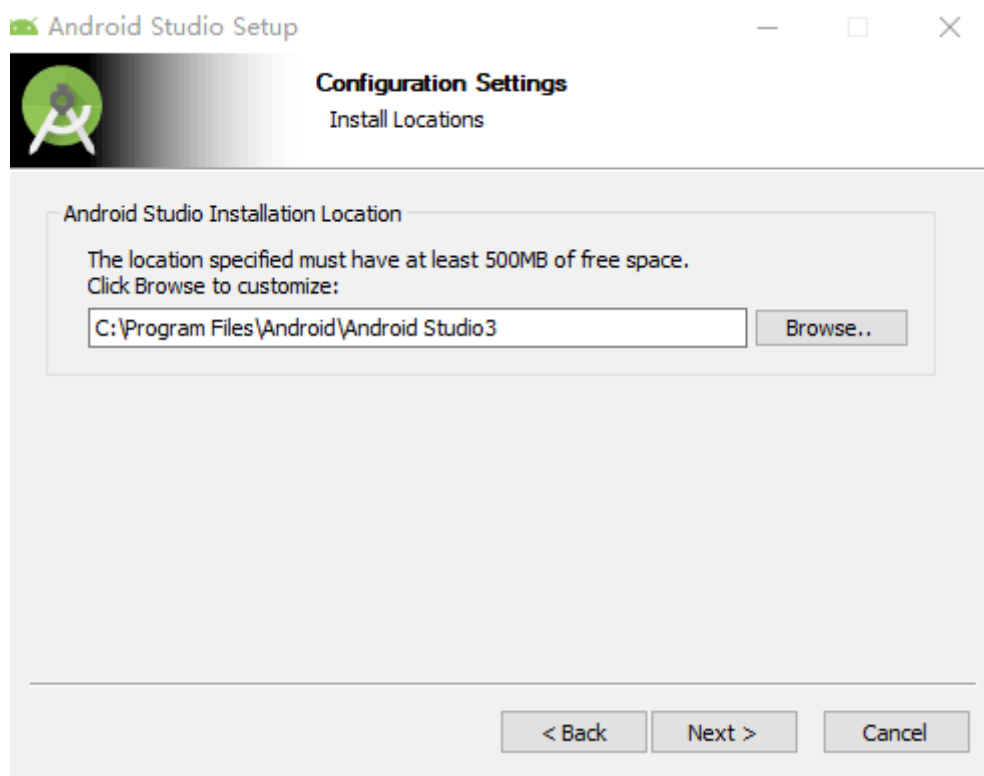


图 1-14 选择安装位置

(3) 选择创建开始菜单图标,如图 1-15 所示。

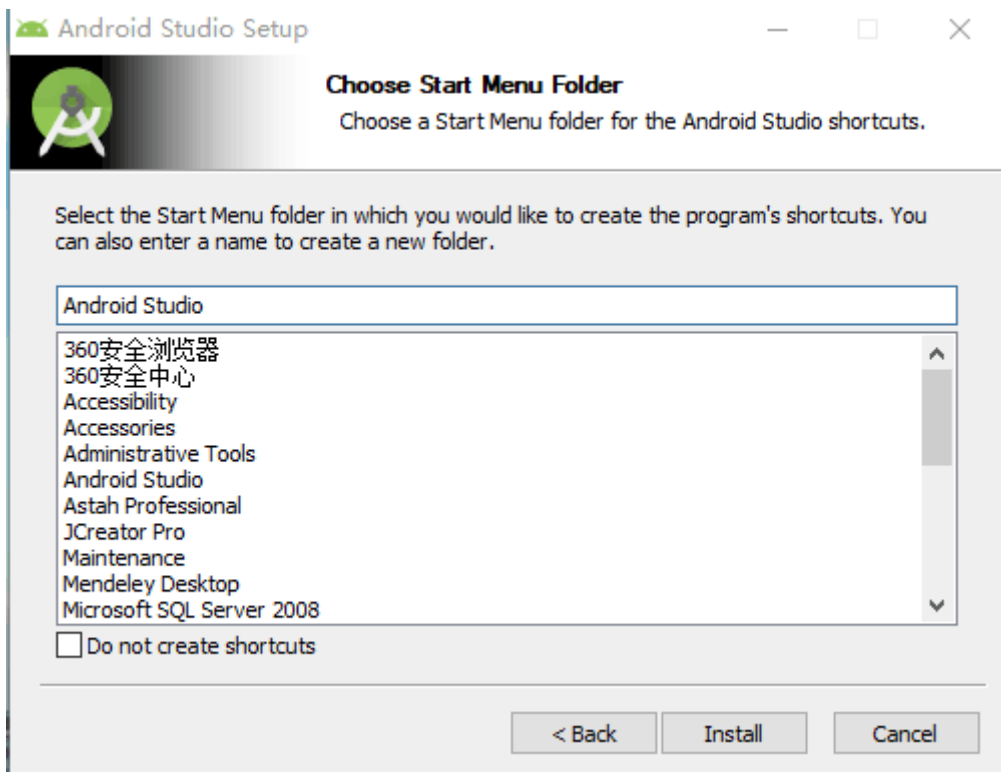


图 1-15 创建开始菜单图标

(4) 下一步需要在线安装，如图 1-16 所示。在线安装的速度可能会很慢，需要耐心等待。

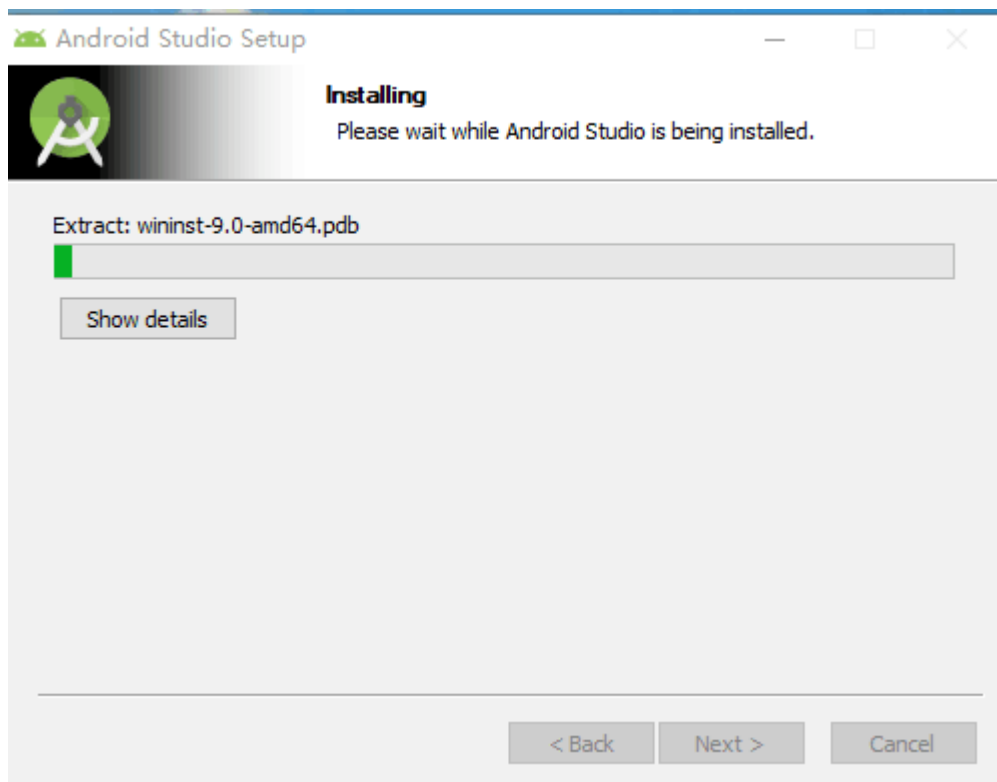


图 1-16 开始安装

(5) 安装完成，如图 1-17 所示。

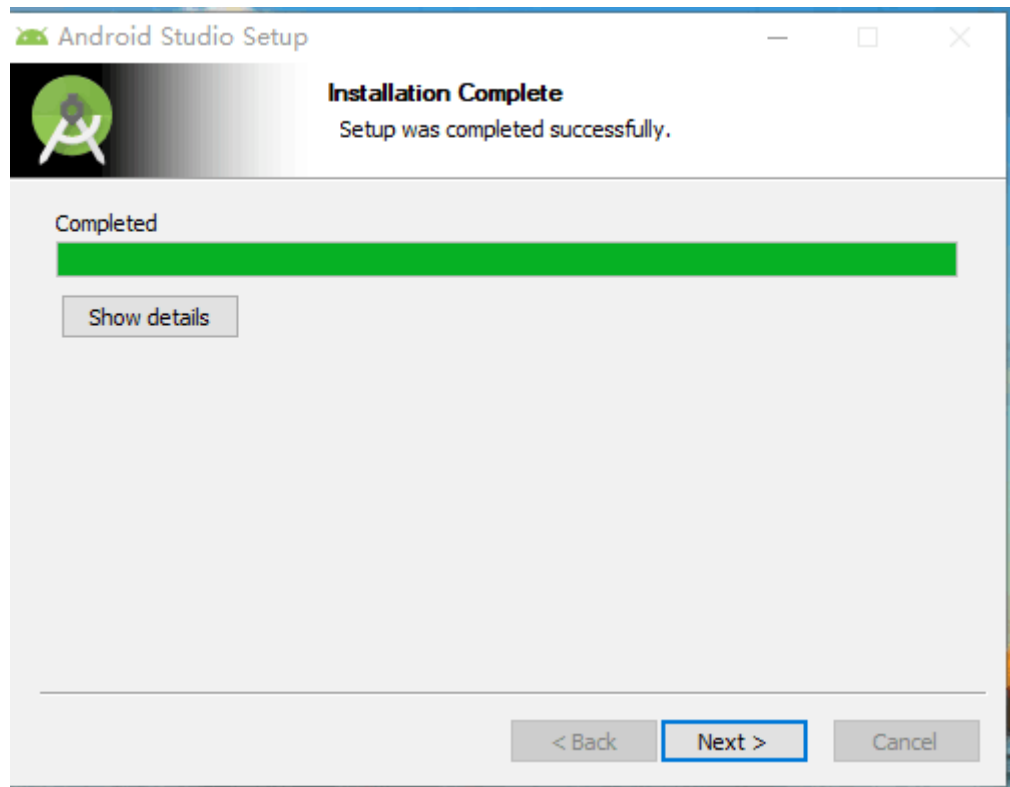


图 1-17 安装完成界面

如果没出什么意外，一小段时间后你就会看到图 1-18 界面，也就说明安装成功了。安装注意事项：

- 安装目录不允许出现中文。
- Android studio 安装前提是 JDK 安装成功，务必保证 JDK 与 SDK 安装成功。

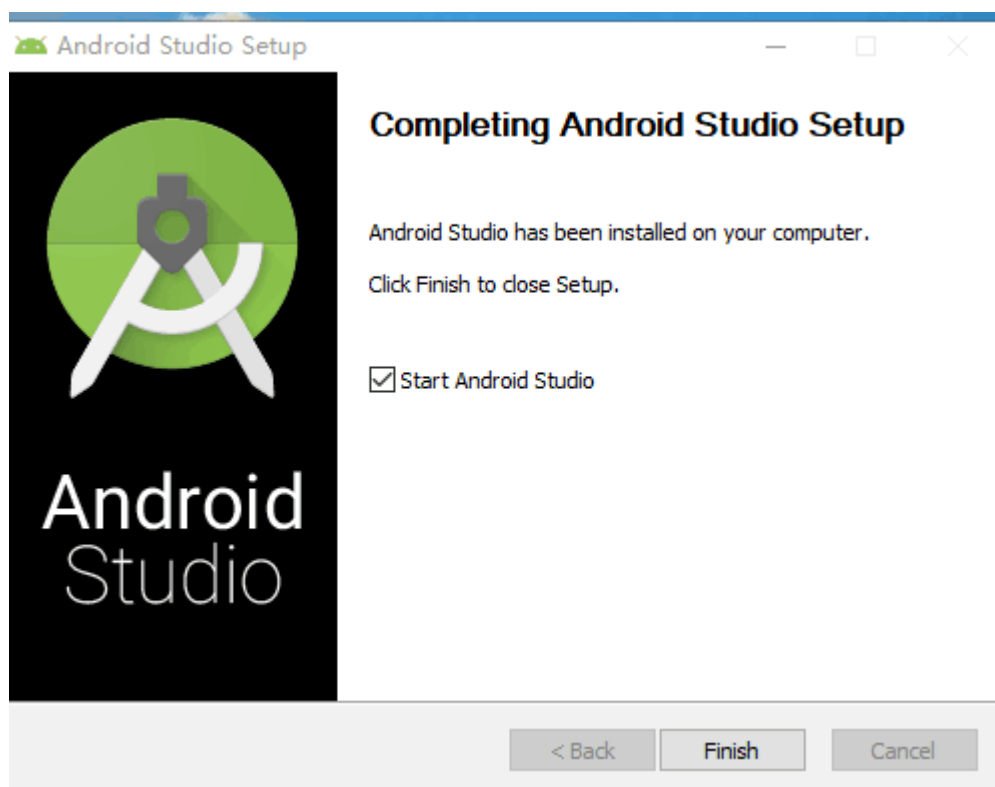


图 1-18 完成安装

1.3 模拟器安装创建

只是有了开发环境是无法测试 Android 程序的，还需要配置 Android 的测试环境，包括模拟器和真机测试环境。

1.3.1 Android 自带模拟器创建

在工具栏上单击  按钮，启动 Device Manager 对话框，如图 1-19 所示。

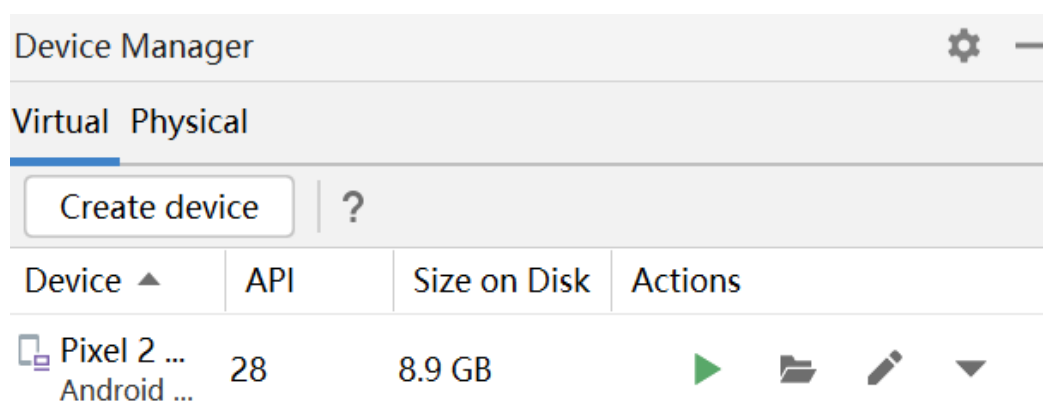


图 1-19 模拟器管理页面

点击“Create Device”按钮按如下步骤设置。

1. 选择机型，如图 1-20 所示。

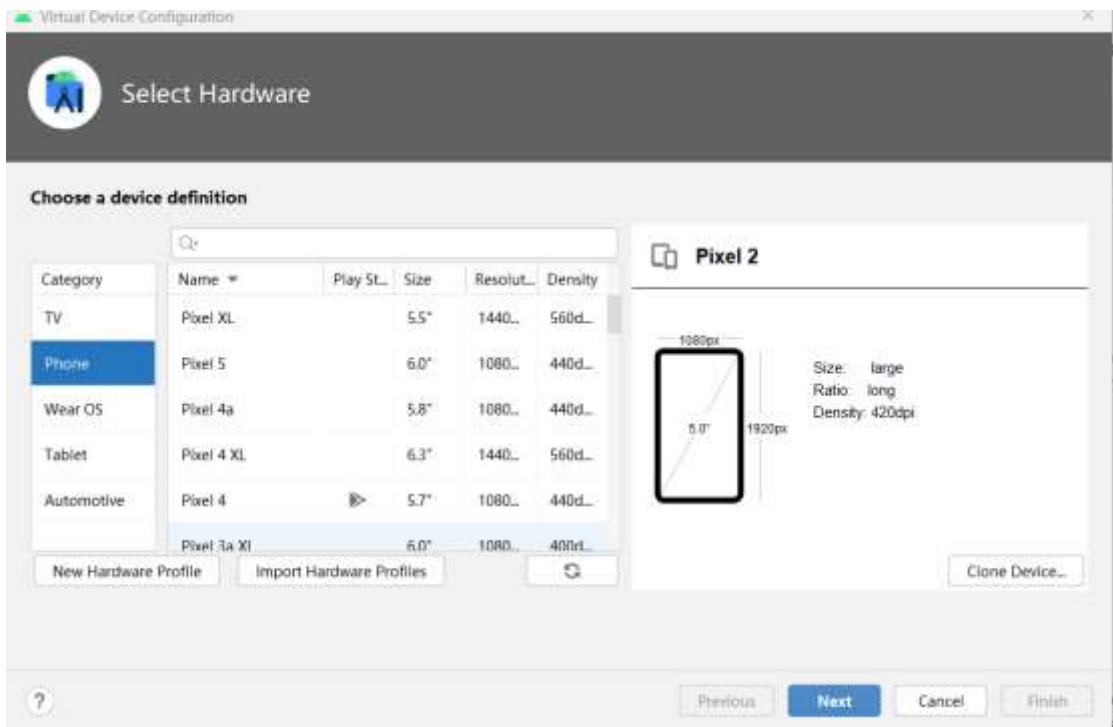


图 1-20 选择机型

2. 选择操作系统，如图 1-21 所示。

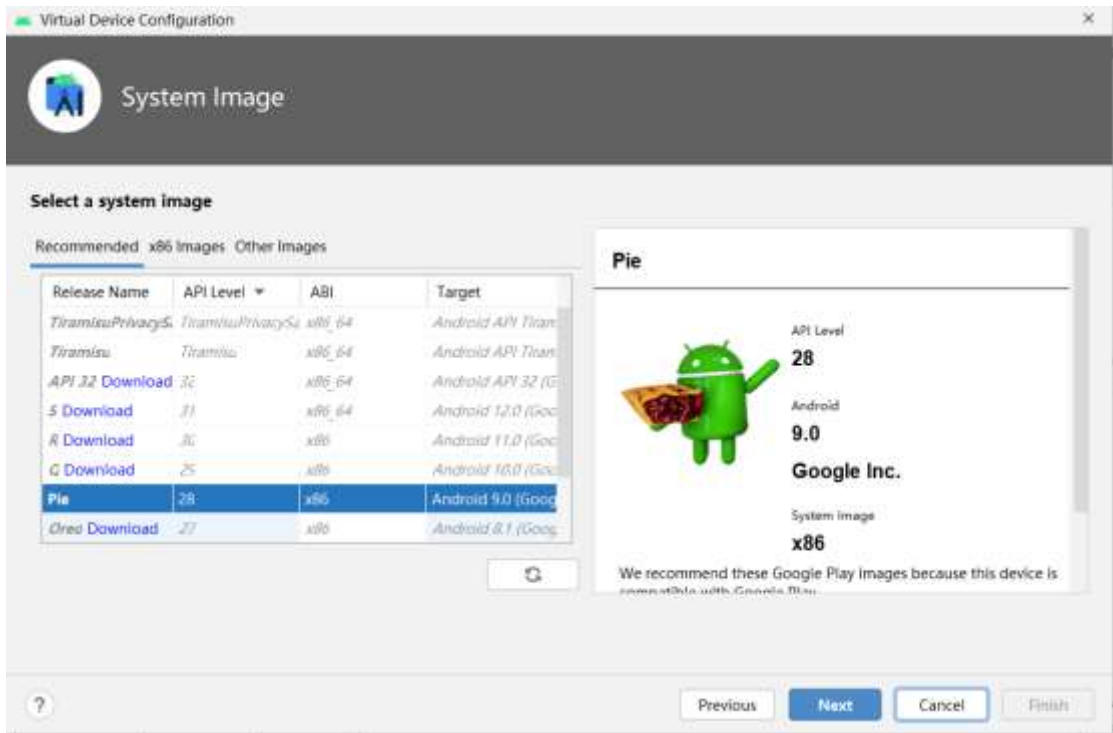


图 1-21 选择版本

3 配置保持默认，如图 1-22 所示。



图 1-22 确认配置

1.3.2 使用真机运行 Android 程序

尽管在 Android 模拟器中可以测试大多数 Android 应用程序，但是 Android 模拟器运行的速度非常慢（与真机相比），而且将 Apk 安装到 Android 模拟器上的过程也非常慢，而且模拟器无法测试使用蓝牙、传感器、NFC 等技术的程序。因此，建议读者尽量使用真机进行测试。

使用真机进行测试首先要用 USB 线将手机与 PC 相连。若是 Ubuntu Linux 和 Mac OS X 在不用安装其他驱动的情况下可以检测到几乎所有的 Android 手机和平板电脑。但如果是 Windows，基本上检测不到任何 Android 真机设备，所以要为特定的 Android 手机安装相应的驱动程序。不过很多读者不知道上哪下载这些驱动，建议利用一些手机管理软件自动的驱动让 Windows 识别 Android 手机或者平板电脑。

现在的手机管理软件很多，例如 91 助手、360 手机助手等均可。安装完手机助手后，将手机用 USB 线与 PC 连接，并运行手机助手，第一次运行会安装驱动程序，安装完后，如果手机助手可以检测到手机，那么 Android 开发环境就可以检测到手机。使用真机测试程序，同时注意需要把手机的“开发者选项”打开，不同手机开发者选项打开的方法略有区别，读者可以查找自己手机打开的方法。

1.4 快速创建一个项目验证环境

1.4.1 创建第一个 Android 项目-Hello World 项目

创建第一个 HelloWorld 项目步骤如下。

1. 创建工程。第一次启动 Android Studio，会出现如图 1-23 所示界面。此界面几个选项从上到下顺序依次如下：

选项 1：创建一个 Android Studio 项目。

选项 2：打开一个 Android Studio 项目。

选项 3：从版本控制系统中导入代码。支持 CVS、SVN、Git、Mercurial，甚至 GitHub。

选项 4：分析和调试 APK。

选项 5：导入非 Android Studio 项目。比如原生的 Eclipse Android 项目，IDEA Android 项目。如果你的 Eclipse 项目使用官方建议导出（即使用 Generate Gradle build files 的方式导出），建议使用选项 2 导入。

选项 6：导入官方样例，会从网络上下载代码。此功能在以前的测试版本中是没有的，建议多看一看官方给的范例。

Configure：设置。

Get Help：帮助文档。

这里选择第一项。

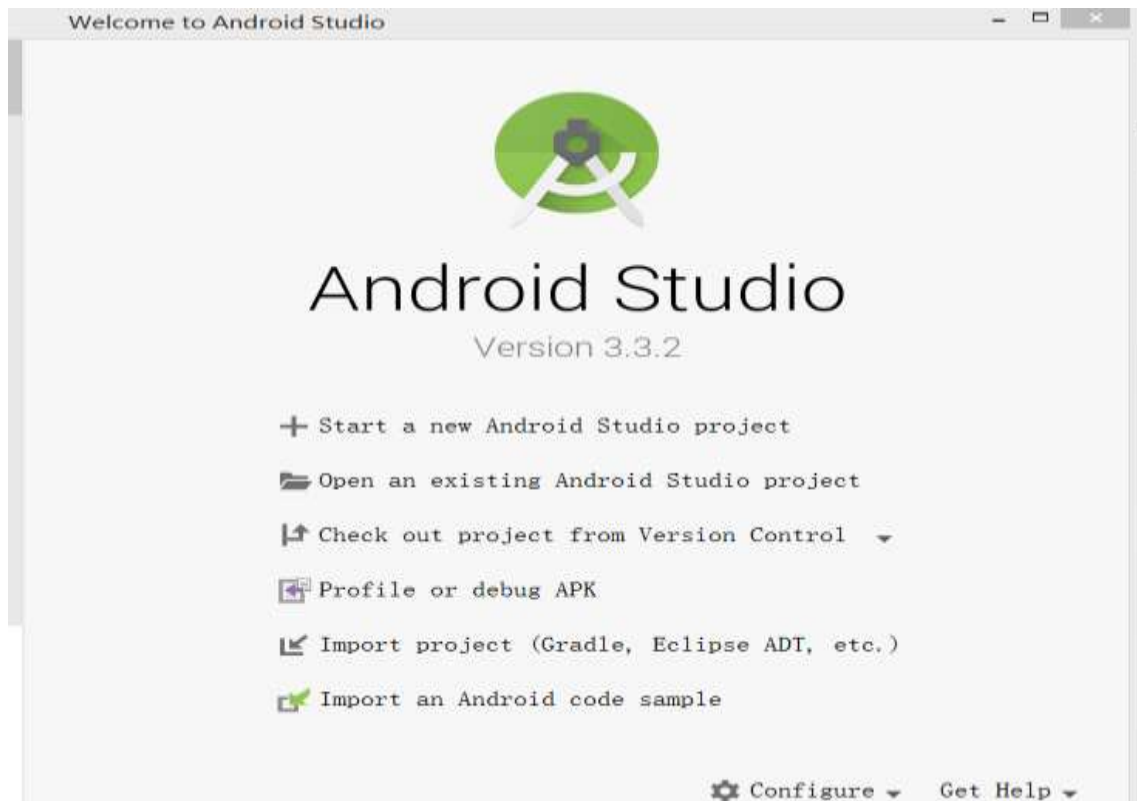


图 1-23 启动页面

2. 选择工程类型，这里选择“Phone and Tablet”，如图 1-24 所示，表示针对手机和平板类型的项目，“Wear OS”表示穿戴设备，“TV”表示电视，“Android auto”表示自动驾驶，“Android Things”表示 Android 物联网。

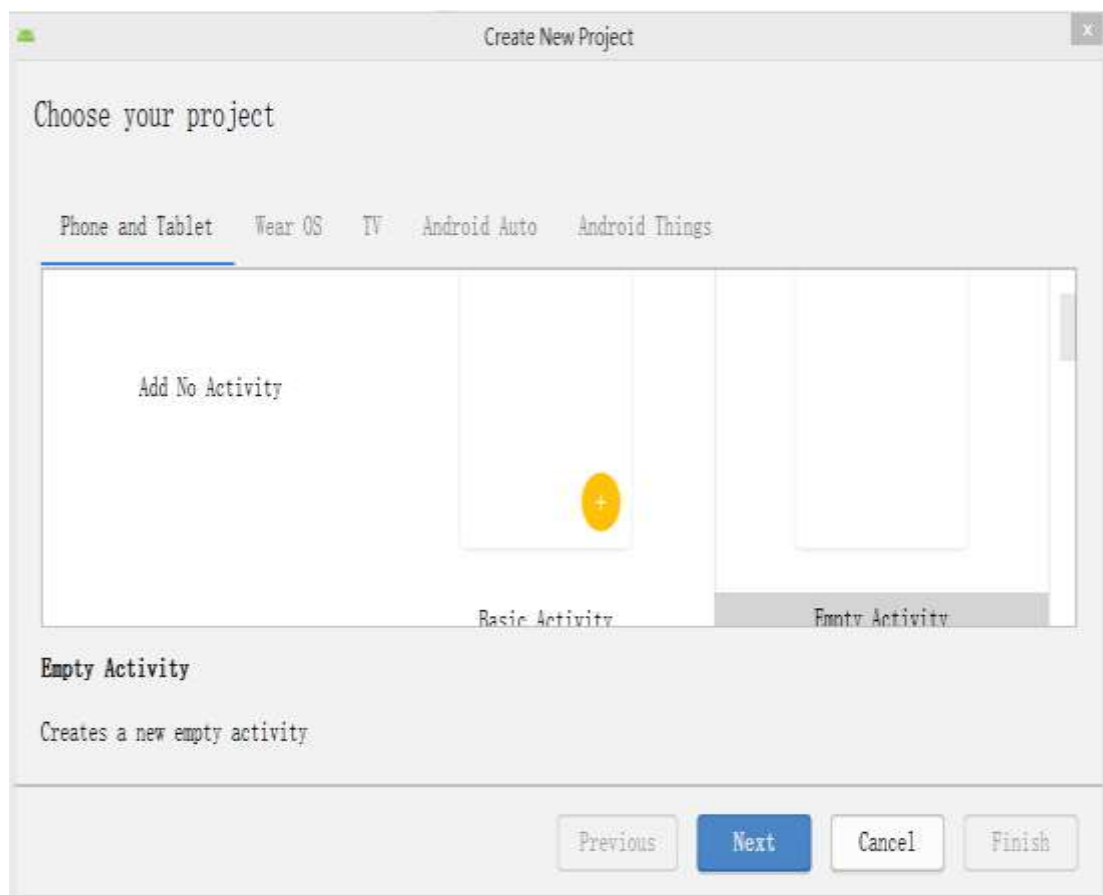


图 1-24 选择工程类型

3. 配置工程，设置工程名称、包名、存放位置以及使用的语言，如图 1-25 所示。

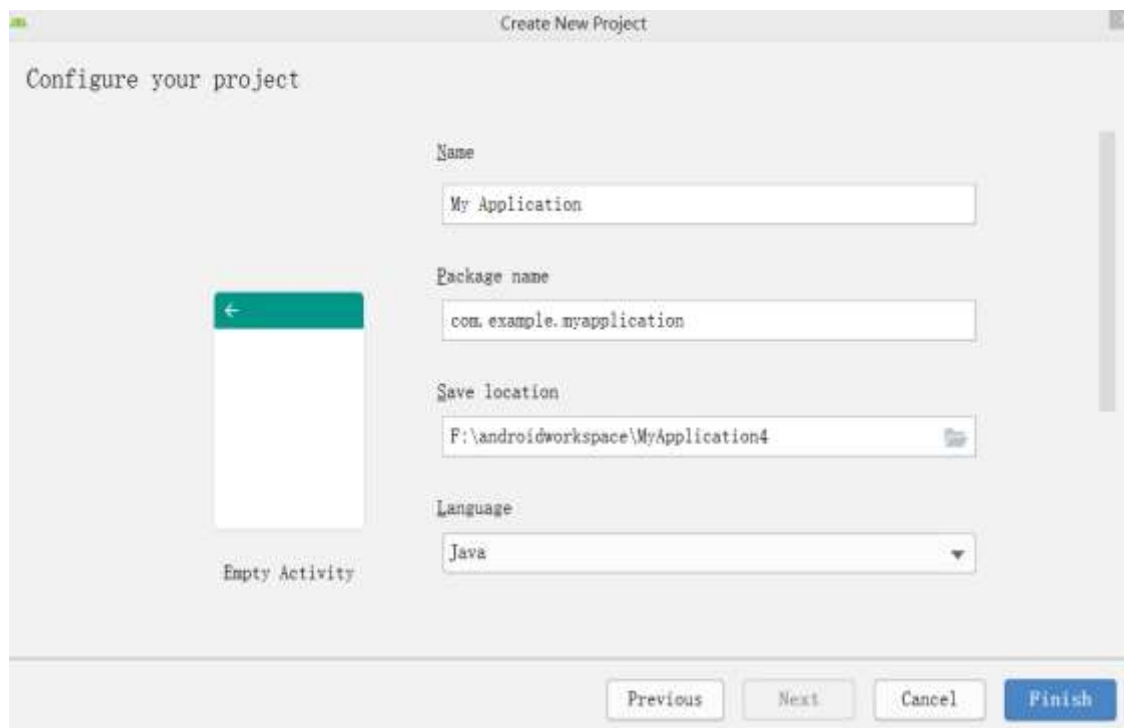


图 1-25 配置工程

点击 finish，使用模拟器运行该项目，其效果如图 1-26 所示。

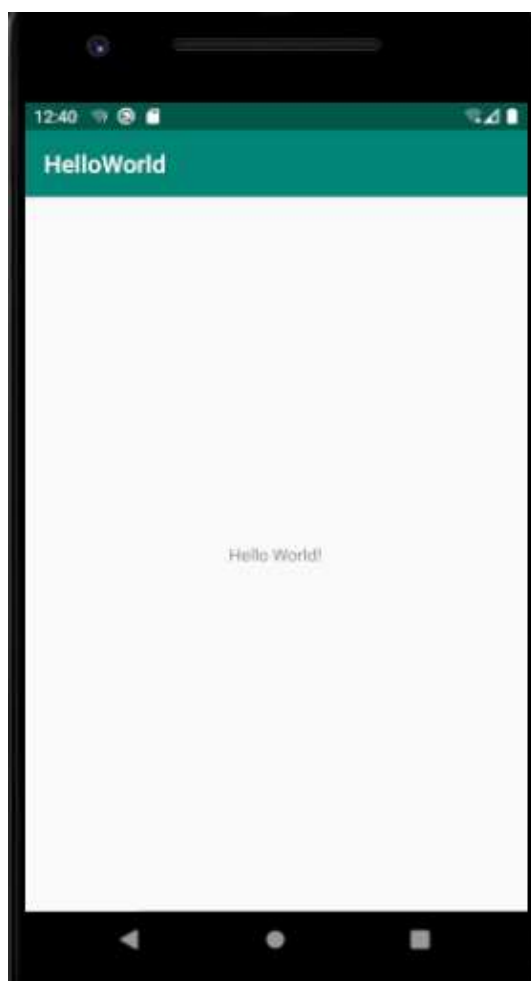


图 1-26 程序运行结果

创建好的工程里默认包含一个模块，可以在一个工程中创建多个模块，创建方法如下，点击 File->new->New Module,弹出如下对话框，选择第一项，如图 1-27 所示。

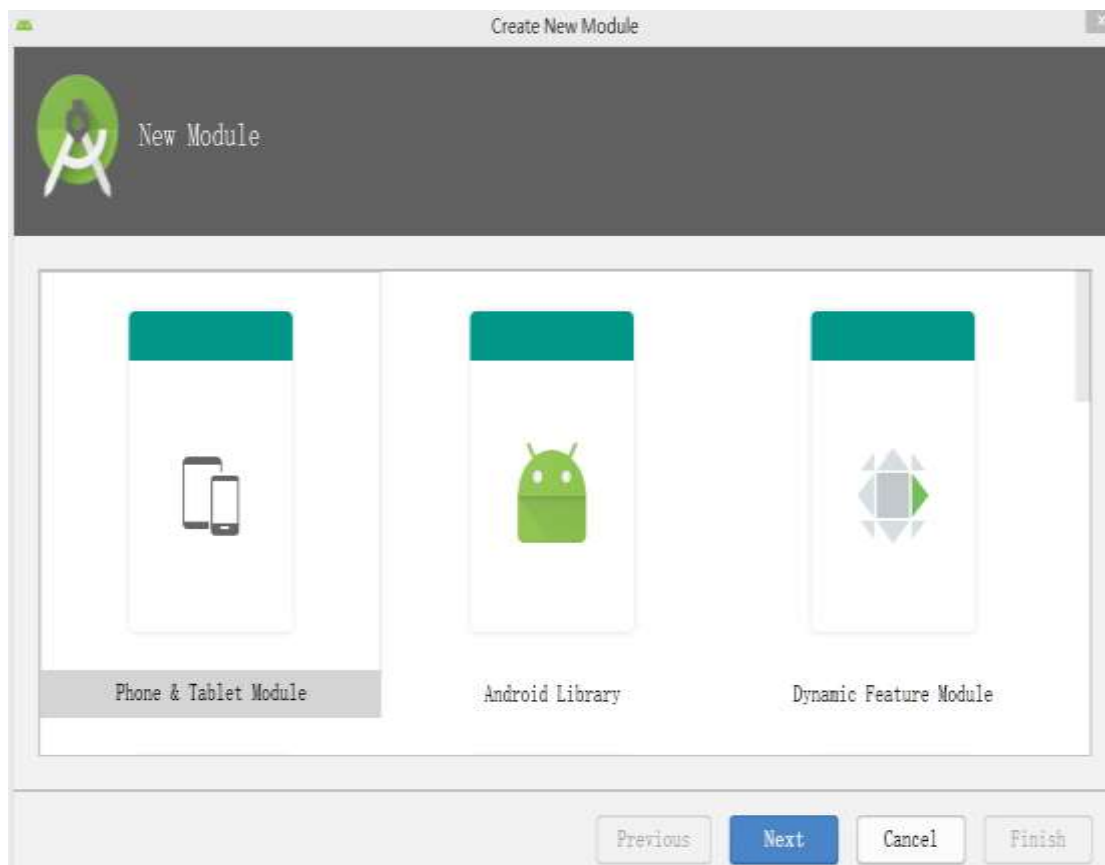


图 1-27 配置工程

填写模块名称，如图 1-28 所示。

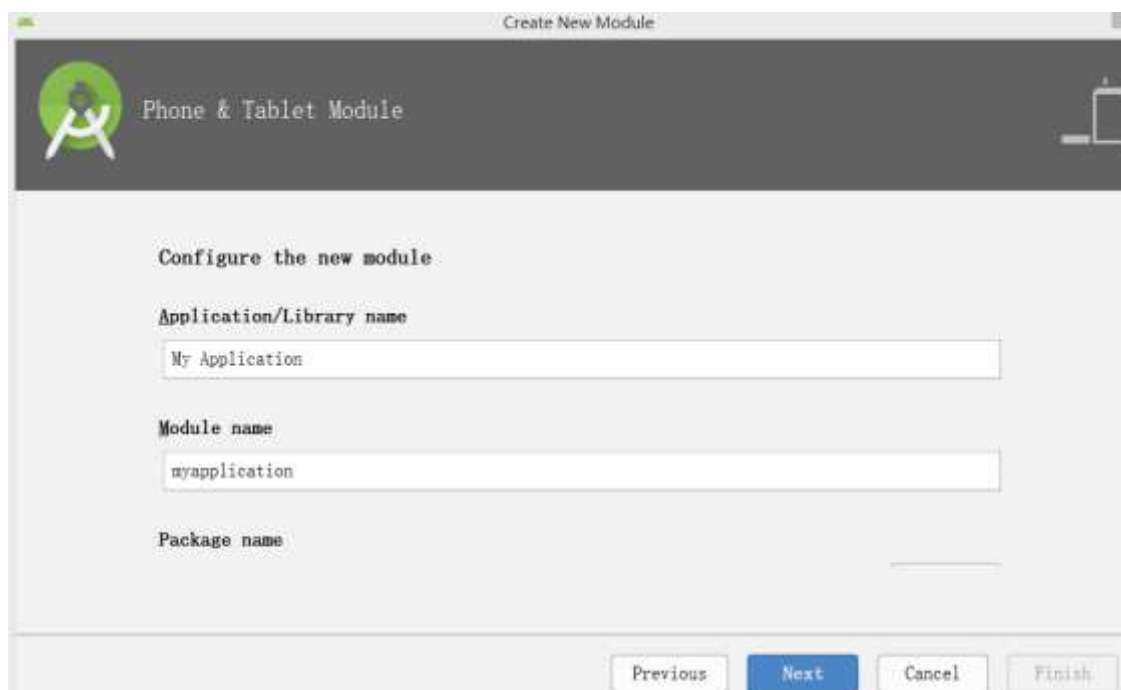


图 1-28 配置工程

选择 Activity 的类型，Activity 是 Android 应用程序的基本功能单元，通常用来与用户交互，界面如图 1-29 所示。

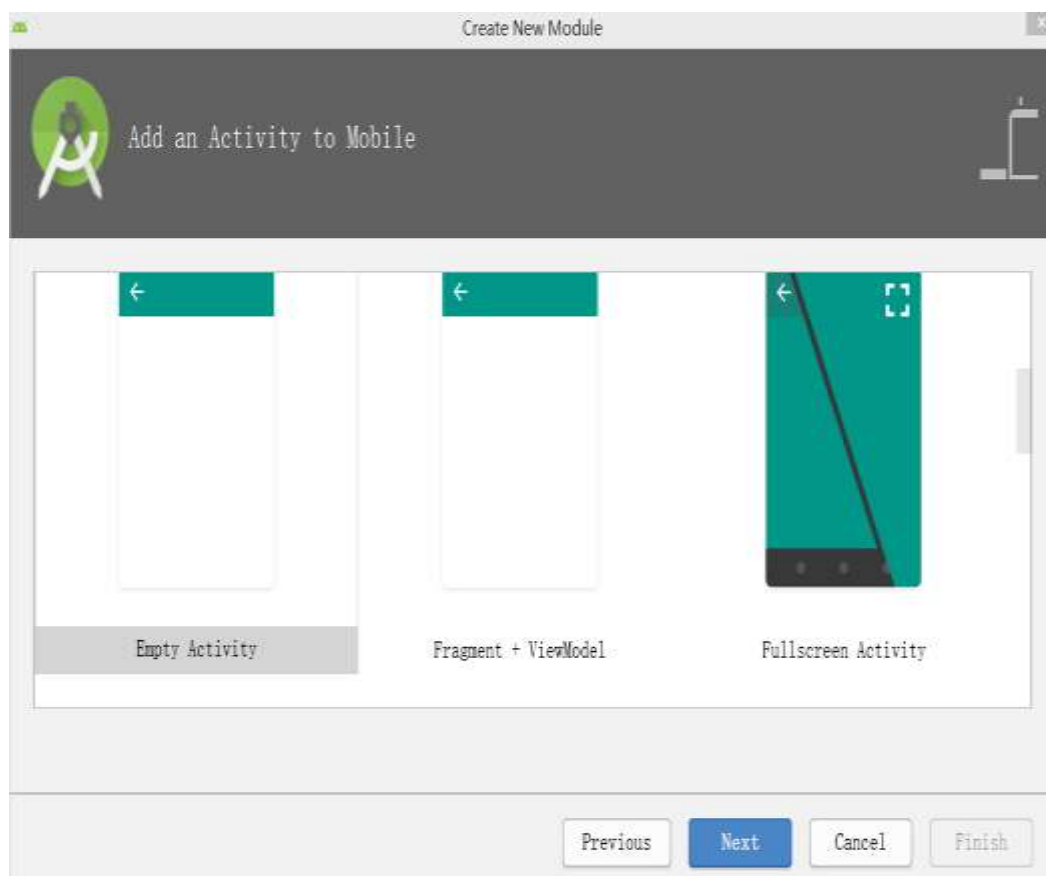


图 1-29 选择 Activity 的类型

这个页面选择一个 Activity 模板，直接选择一个 Empty Activity，进入如图 1-30 所示界面。

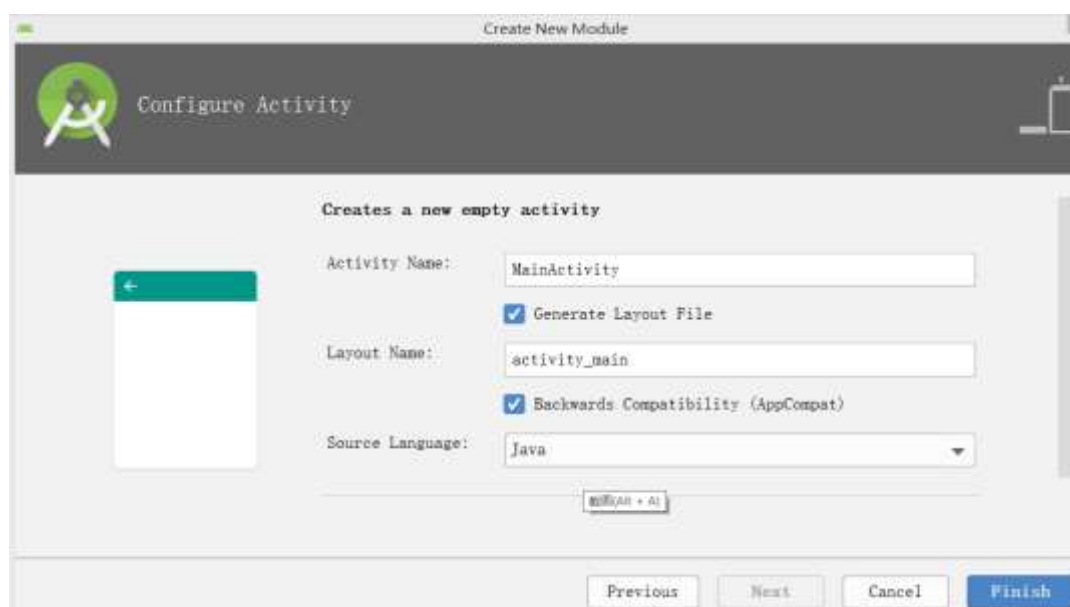


图 1-30 设置 Activity 和布局文件

1.4.2 Android 项目各目录与文件作用分析

在 Android Studio 中，提供了如图 1-31 所示几种项目结构类型：

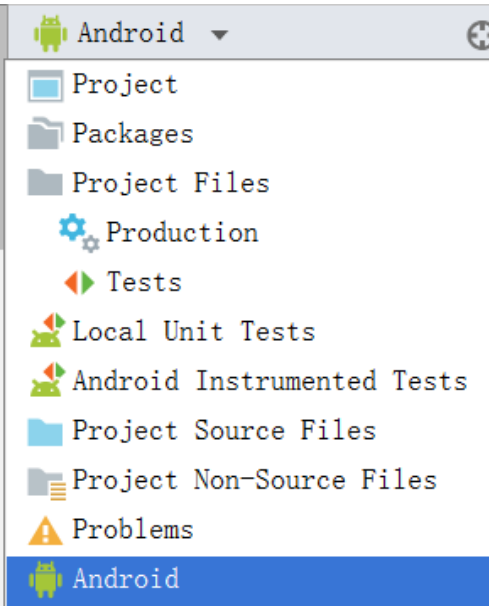


图 1-31 项目结构类型

一般常用的有以下两种结构：Project 结构类型和 Android 结构类型。Project 结构类型如图 1-32 所示。

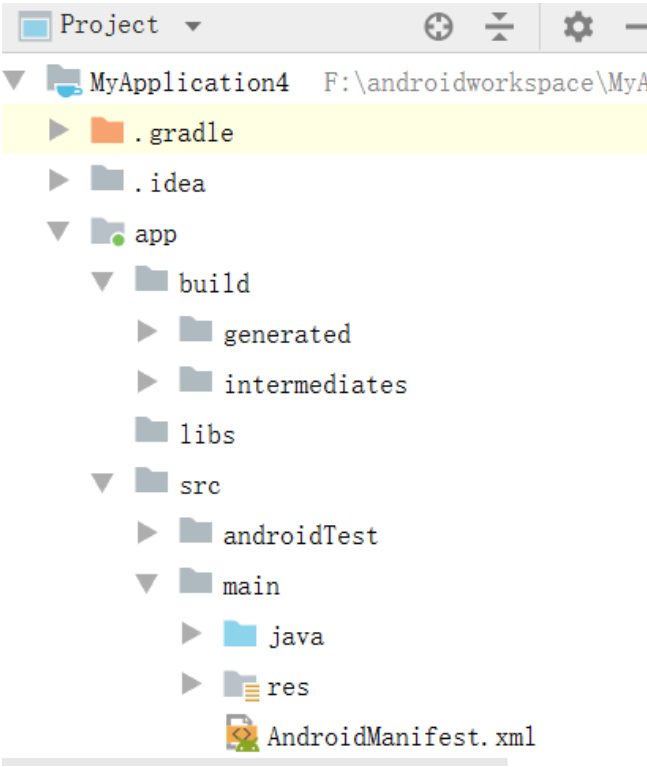


图 1-32 Project 结构类型目录结构

Android 结构类型如图 1-33 所示。

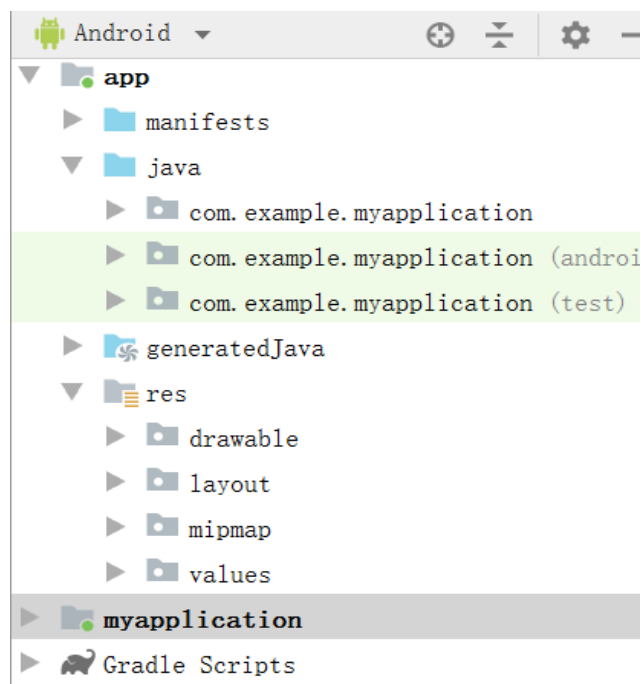


图 1-33 Android 结构类型目录结构

各目录功能结构说明如下：

1. app/manifests AndroidManifest.xml 配置文件目录。相当于一个部署文件，指明应用所在的包，应用的图标、对 Activity、Service、broadcast 进行声明注册之后，通过 intent-filter 意图过滤器就能实现组件之间的切换，还需设置应用的权限、版本。

AndroidManifest.xml 文件包含如下元素

(1) Manifest:

是根节点，描述了 package 中所有的内容。

(2) xmlns:android:

定义 android 命名空间，一般 <http://schemas.android.com/apk/res/android>，这样使得 Android 中各种标准属性能在文件中使用，提供了大部分元素中的数据。

(3) package:

指定本应用内 java 主程序包的包名，它也是一个应用进程的默认名称。

(4) android:icon

这个很简单，就是声明整个 APP 的图标，图片一般都放在 drawable 文件夹下。

(5) android:label

这个属性用于给内容提供者定义一个用户可读的标签。如果这个属性没有设置，那么它会使用<application>元素的 label 属性值来代替。

(6) <Application>

一个 AndroidManifest.xml 中必须含有一个 Application 标签，这个标签声明了每一个应用程序的组件及其属性(如 icon、label、permission 等)。

(7) <intent-filter>

这个元素用于指定 Activity、Service 或 Broadcast Receiver 能够响应的 Intent 对象的类型。Intent 过滤器声明了它的父组件的能力 Activity 或 Service 所能做的事情和 Broadcast Receiver 所能够处理的广播类型。它会打开组件来接收它所声明类型的 Intent 对象,过滤掉那些对组件没有意义的 Intent 对象请求。过滤器的大多数内容是通过<action>、<category>和<data>子元素来描述的。

(8) Action:组件支持的 Intent action。

(9) Category:组件支持的 Intent Category。

2. app/java 源码目录。这里面是写的普通 Java 类，它继承于 Activity 的类，即项目源代码。如果工程比较复杂，可以分成许多包，每个包中可以有很多类，这样可以使代码结构清晰，便于开发阅读。

3. app/res 资源文件目录。存放应用程序用到的资源文件，其包含 anim, drawable, layout, values, raw 等目录。当这个文件下的文件发生变化时，R 会自动发生变化。

4. Gradle Scripts gradle 编译相关的脚本。

5. build----编译后的文件存放的位置,最终 apk 文件存放的位置

6. libs:存放.jar 和.so 文件

7. src:AndroidTest 与 test 存放测试相关的内容

1.5 课后实践作业

1. 简要说明一个 Android 工程的结构及 AndroidManifest.xml 的作用。

/src 目录

/res 目录

/libs 目录

AndroidManifest.xml

2. 如何使用真机测试 Android 程序?

3. 创建一个 Pixel 2 API 29 的模拟器。

4. 完成安装环境，能够运行出 HelloWorld 项目

