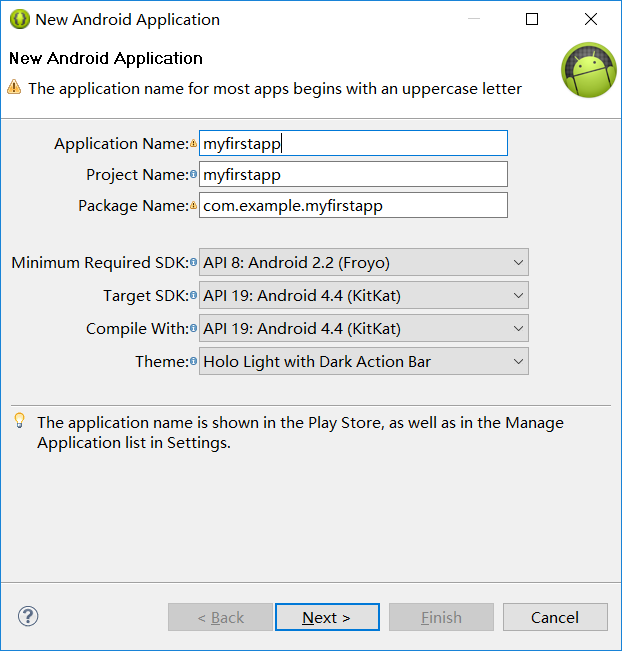
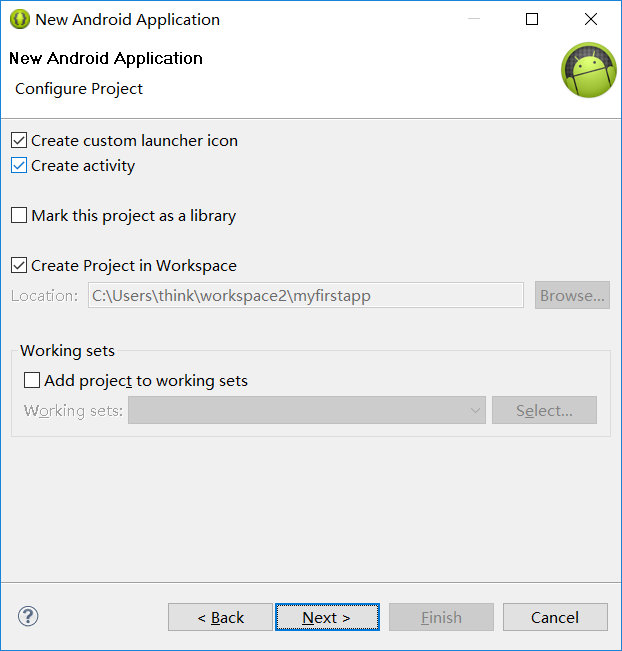
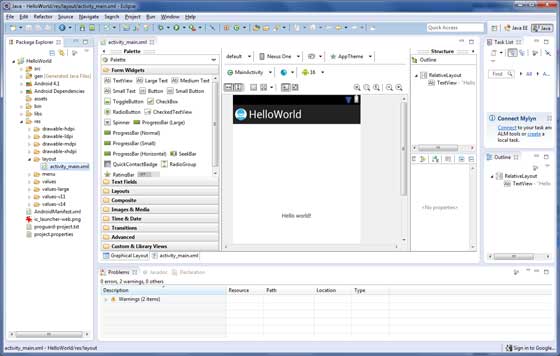
**创建 Android 应用**

当我们配置好开发环境后。第一步是通过Eclipse IDE来创建一个简单地Android应用程序。按照选项File -> New -> Project，最后从向导列表选择 Android New Application 。现在，使用如下的窗口向导将应用程序命名为HelloWorld：

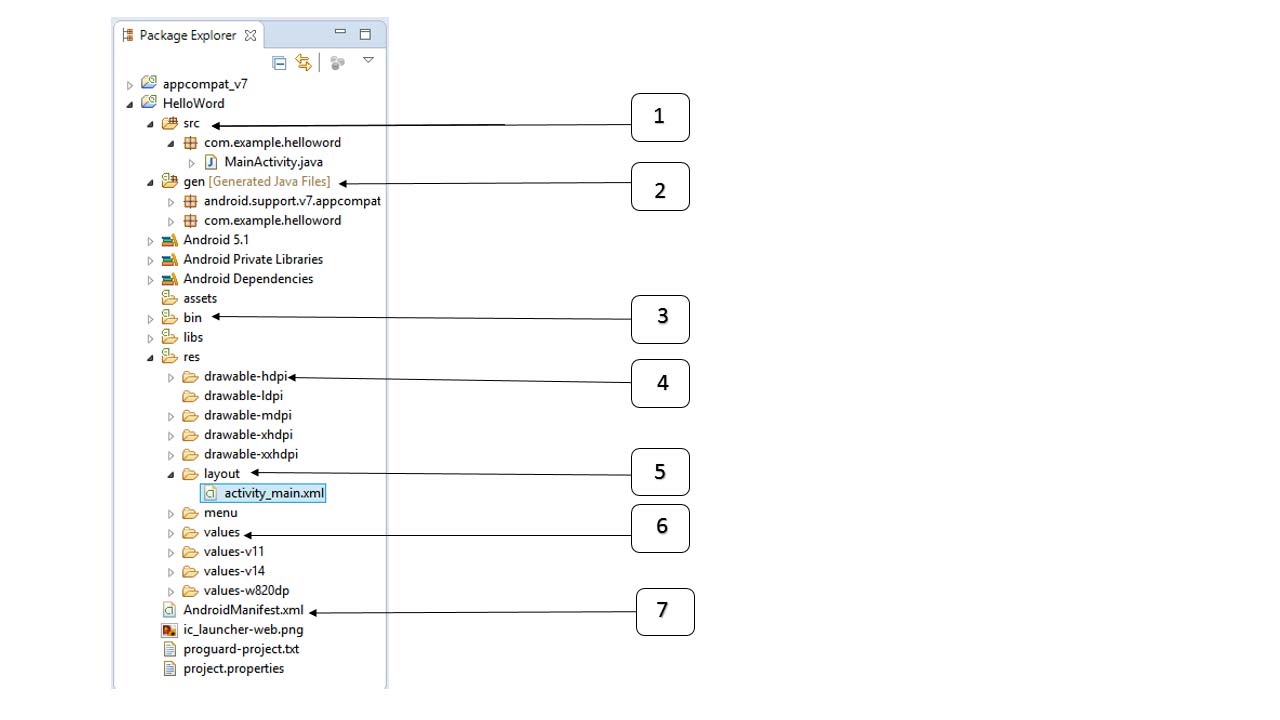




这里保持第二个选项，create activity选项的对勾不要去掉。让工程自动创建主Activity。保持所有的默认输入，直到最后一步。一旦项目创建成功，将会看到如下的项目界面：



**安卓Eclipse工程组织**



| **序号** | **文件夹、文件和说明** |
| --- | --- |
| 1 | src:包含项目中所有的.java源文件，默认情况下，它包括一个 MainActivity.java源文件对应的活动类，当应用程序通过应用图标启动时，将运行它。 |
| 2 | gen:这包含由编译器生成的.R文件，引用了所有项目中的资源。该文件不能被修改。 |
| 3 | bin:这个文件夹包含Android由APT构建的.apk包文件，以及运行Android应用程序所需要的其他所有东西。 |
| 4 | res/drawable-hdpi:这个目录下包括所有的为高密度屏幕设计所需的drawable对象。 |
| 5 | res/layout:这个目录存放用于定义用户界面的文件。 |
| 6 | res/values:这个目录存放各种各样的包含一系列资源的XML文件，比如字符串和颜色的定义。 |
| 7 | AndroidManifest.xml:这个是应用程序的清单文件，描述了应用程序的基础特性，定义它的各种组件。 |

**一些重要的文件**

**主界面的java源程序**：这是实际的应用程序文件，将被转化为Dalvik可执行文件并运行。下面是由应用向导为Hello World应用生成的默认代码。

**清单文件**：需要在应用程序项目根目录下的manifest.xml文件中声明所有的组件。这个文件是Android操作系统与当前APP的应用程序之间的接口，因此，如果没有在这个文件中声明的组件，将无法被操作系统所识别。

**Strings 文件**：Strings.xml文件在res/value文件夹下，它包含应用程序使用到的大部分的文本。例如，按钮、标签的名称，默认文本等等。虽然并不是必须的但是建议将配置文件（XML）中的文本都集中在这里，方便修改或扩展。

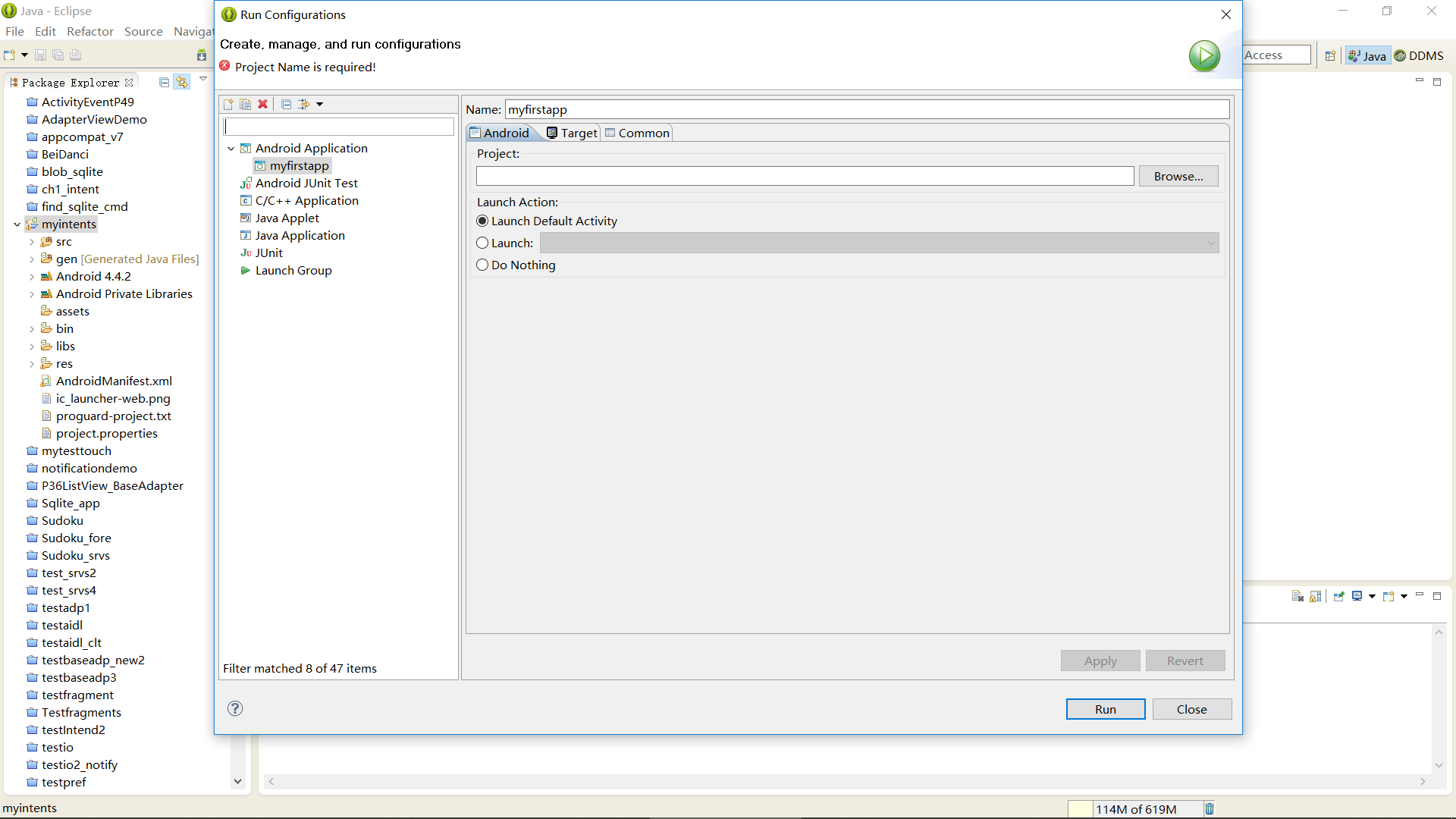
**R 文件**：gen/com.example.helloworld/R.java文件是活动（自动生成/修改）的Java文件，为工程中的.java源程序提供资源XML配置文件中各种资源的标识（一一对应的整数值）。

**Layout 文件**: activity\_main.xml是一个在res/layout目录下的layout文件。当应用程序构建它的界面时被引用。开发者将非常频繁的创建或修改这些布局文件来设计APP的界面。在"Hello World"应用程序中，这个文件自动生成且具有默认的布局。

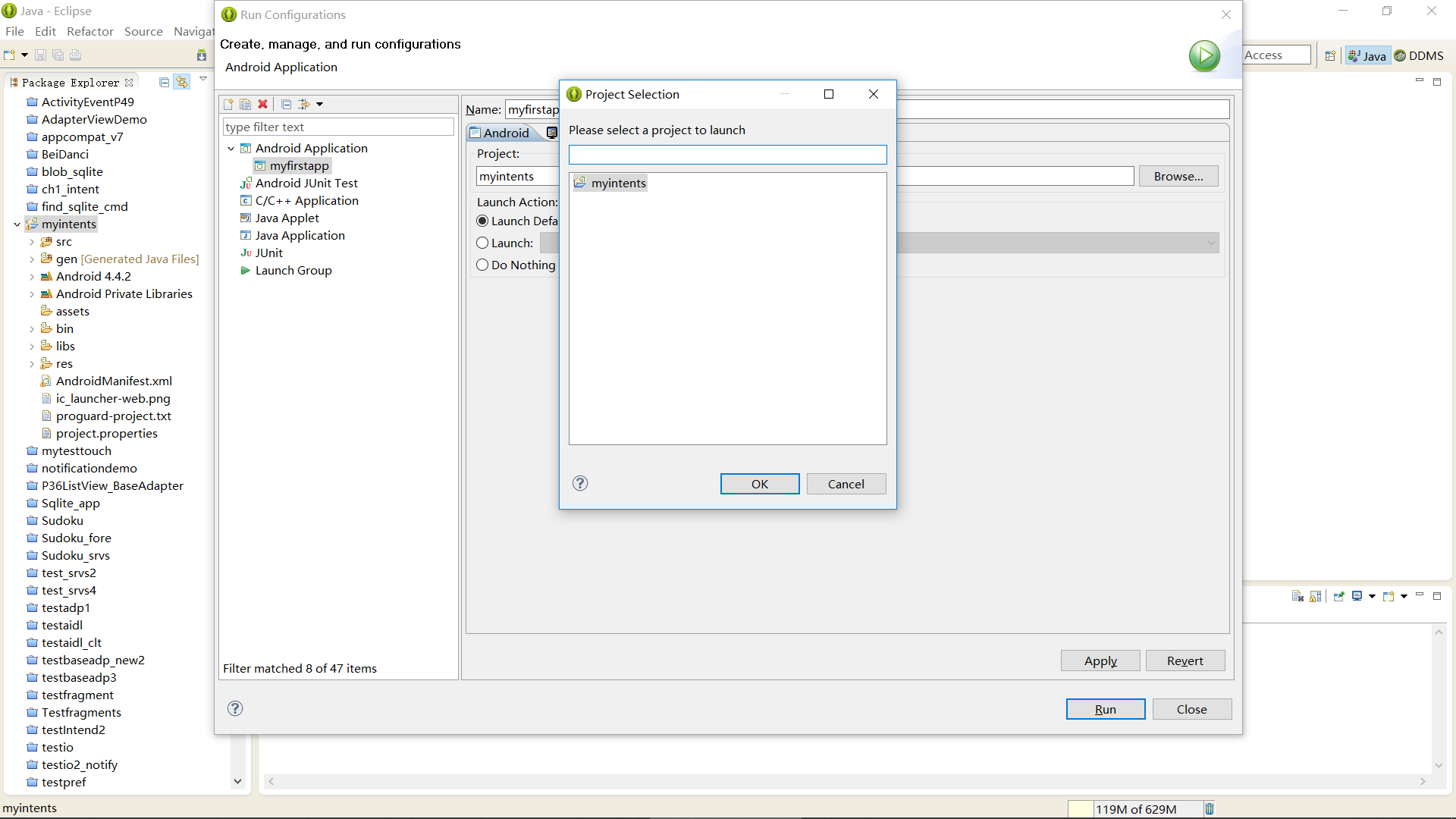
**运行**

首先确定已经创建好一个安卓虚拟机或连接到打开开发者选项的安卓设备上。

进入run configuration，并创建一个android application的运行实例，建议修改运行实例的名称。



通过文件浏览确定该运行实例对应的工程。



此时target标签页中应该可以看到备选的虚拟机或联机设备。可以在这里选择，或者直接运行run再选择。最后运行成功看到运行效果。类似于：

